Efficient Simulation via Validation and Application of an External Analytical Model

DISSERTATION

Thomas H. Irish
Major, USAF

AFIT/DS/ENS/99-01

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/DS/ENS/99-01

Efficient Simulation via Validation and Application of an External Analytical Model

DISSERTATION

Thomas H. Irish
Major, USAF

AFIT/DS/ENS/99-01

The views expressed in this dissertation are those of the author and do not reflect the official policy or position of the Department of Defense or the United States Government.

AFIT/DS/ENS/99-01

Efficient Simulation via Validation
and Application of an
External Analytical Model

DISSERTATION

Presented to the Faculty of the Graduate School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy in Operations Research

Thomas H. Irish, B.A., M.S.
Major, USAF

September 14, 1999

Approved for public release; distribution unlimited

AFIT/DS/ENS/99-01

Efficient Simulation via Validation and Application of an External Analytical Model

Thomas H. Irish, B.A., M.S.

Major, USAF

Approved:

Kenneth W Bauer Jr. — 1 SEP 99

Dr. Kenneth W. Bauer Jr. — Date
Committee Chair

1 Sep 99

Lt Col John O. Miller — Date
Committee Member

Richard A. Raines — 1 Sep 99

Maj Richard A. Raines — Date
Committee Member

Dennis W Quinn — 1 SEP 99

Dr. Dennis W. Quinn — Date
Dean's Representative

Accepted:

Robert A. Calico, Jr.
Dean, School of Engineering

Acknowledgements

I would not have completed this dissertation without the support of many people.

I owe a large debt of gratitude to my first research advisor, Lt Col Dennis Dietz (ret). It was his idea of performing research into analytical control variates that got me to this point. Although he was unable to complete the process with me, I still consider him one of my committee chairs.

The members of my committee, Lt Col J.O. Miller and Maj Richard Raines provided me with support and encouragement throughout the entire process.

I really don't know how to thank my final committee chair, Dr Ken Bauer, enough. I never doubted that he sincerely wanted to see me do my best and graduate. I believe he is truly motivated by his desire to teach and to help his students reach their goals. Thanks Dr Bauer.

Of course words are never enough when it comes to thanking your family. I learned what family means from my parents, Don and Carol Lee. They always put me and my brothers and sister first. And they always had faith in each of us, no matter what. They still do. I'd like to think that there is a lot of them in me.

My boys, Kevin and Sean, are great. They always make me so proud. They provided me with the inspiration to keep on trying even when I thought I didn't want to.

My wife Gloria means the world to me. She sure had to put up with a lot. And she did. I wasn't always the easiest person to live with the last three and half years. Yet she still loves me. And I love her. The same as I did that morning in the Bomb-Nav shop when she first entered my heart.

Thomas H. Irish

Table of Contents

List of Figures

Figure Page

List of Tables

Table Page

Table Page

AFIT/DS/ENS/99-01

Abstract

This dissertation research makes significant contributions towards the synergistic use of both analytical and simulation models for improving the efficiency of simulation studies. The foundation for this research is the application of the analytical control variate (ACV) method. The ACV method employs an external analytical model to consolidate multiple input random variables into a single ACV. Previous research suggests that this approach can produce significant variance reduction, but the resulting point estimate of the simulation response may exhibit unacceptable bias. In this research a general Monte Carlo sampling method for resolving the bias problem is developed and demonstrated through a queueing network example. In order to use the method, the means, variances, and approximate distributions of the random variables used to produce the ACV must be known.

For some simulation models, not all of the means, variances, and approximate distributions of the random variables used to produce the ACV are known. In this research both parametric and non-parametric alternatives to the Monte Carlo method are explored for these cases. The effectiveness of these methods is demonstrated using an airfield simulation model.

Significant variance reduction using an ACV indicates that the outputs of both models are highly correlated when subjected to similar inputs. This relationship is exploited in this research and a new methodology is developed for conducting searches of a simulation design space using an analytical model vice a simulation model. The justification for the new *surrogate search* method is based on validating the analytical model to the simulation model using techniques adapted from simulation model validation and verification. The validation and surrogate search method are fully integrated within the context of a simulation study by analyzing the results of the ACV method. The effectiveness of the method is demonstrated on two simulation models including the HQ AMC Mobility Analysis Support System (MASS) model.

Efficient Simulation via Validation and Application of an External Analytical Model

I. Introduction

1.1 General Discussion

Air Force and industry analysts use mathematical models to study systems and provide decision-makers with the information necessary to set policy and allocate scarce resources. Examples of mathematical models include simulation, statistical regression, and stochastic analytical. Each of these different types of models has qualities that recommend or discourage their use for a particular system or problem. All three of these types of mathematical modeling are discussed and used in this dissertation, with the focus on simulation and stochastic analytical modeling.

Large, complex systems with stochastic elements are often studied with discrete event simulation models rather than analytical models. The major advantage of using a simulation model is the ability to model system characteristics that currently defy analytical description. The major disadvantage of simulation models is the large amount of time required to complete a sufficiently accurate study. On the other hand, an analytical model can often provide a solution in a relatively short time as compared to a simulation study. Unfortunately, few analytical models exist for large, complex systems since such systems usually have characteristics that do not yield to analytical description. On the other hand, if the system can be solved analytically, the complexity of the system may necessitate solutions that are equally complex with untenable computational or numerical problems. Further, because of the complexity even approximate analytical models are seldom used to study these systems despite the obvious time advantage they can provide. This dissertation makes significant contributions towards the synergistic use of both analytical and simulation models to reduce the time required to complete a simulation study.

Many techniques and methods exist for reducing the time necessary to complete a simulation study including those referred to as variance reduction techniques (VRTs). Due to the nature of discrete event simulation models, several different means of actually reducing the observed variance of performance measure estimators are available to the analyst. By reducing the variance of the estimator, the associated confidence interval is also reduced. Hence, fewer simulation replications are necessary to reach a pre-determined level of accuracy, reducing the time of a study. The method of control variates (CV) is a VRT that takes advantage of the correlation between a CV (a random variable) and the simulation output estimator (another random variable) in order to achieve variance reduction. Depending on the type of CV applied, the correlation might arise naturally during the course of simulation (*internal* control variates) or might be induced by using common random numbers in a separate control simulation model (*external* control variates) [18].

One CV method that combines analytical and simulation models is the analytical control variate (ACV) method [49]. The ACV method uses a separate analytical model to generate a CV that is correlated to the output of a simulation model. To obtain the necessary correlation, the analytical model need only be an approximate representation of either the same underlying system that generated the simulation model or the simulation model itself. Briefly, the ACV method is a hybrid of the two types of typical CV's—internal and external. The moment estimators of the realized input random number streams that drive the simulation model are used (like internal CV's) as inputs to an external (like external CV's) analytical model to generate the ACV. If the analytical model is an adequate representation of the system under study (or the simulation model), the ACV will be sufficiently correlated to the simulation output estimator to produce adequate variance reduction. Unfortunately, previous researchers [48, 49, 53, 54] have all reported unacceptable levels of bias in the ACV controlled estimators. This bias is caused by the necessity to evaluate the expected value of the ACV given the distribution of the input random variables used to produce the ACV [49]. Since the analytical models developed for these stochastic systems are normally non-linear, or even algorithmic in nature, an analytical solution of this model's expected output is extremely difficult if not impossible to generate. We present an efficient solution to the bias problem in this dissertation which is itself a significant contribution to the use of analytical and simulation models in concert to reduce study times.

Now, the fact that the analytical model produces variance reduction suggests that it might be possible to use it in other ways to reduce simulation study times since given similar inputs the two models produce similar outputs. A logical next step is to exploit this relationship and use the analytical model in place of the simulation model when it is shown to be valid for such a purpose. We present an advancement in the area of using analytical and simulation models in an iterative fashion. The newly developed method, called the *surrogate search* method, assumes that the ACV method has produced significant variance reduction so that the analytical model can be used as a surrogate for the simulation model to perform explorations of the simulation experimental design space. Using the analytical model instead of the simulation model to perform this function can save significant amounts of time. We present a general methodology and apply it to a study involving a large-scale simulation model.

1.2 Problem Statement

The sponsor of this research, Air Mobility Command (AMC), uses two discrete event simulation models to analyze their operations—the Base Resource and Airfield Capability Evaluation (BRACE) and the Mobility Analysis Support System (MASS) model. They developed BRACE as an airfield simulation used to estimate an airfield's throughput capacity and resource requirements [1]. BRACE simulates the scheduled flow of aircraft to an airfield with predetermined resources. All the major ground activities each aircraft must accomplish before it can depart the airfield are also simulated. These activities include taxiing, scheduled and unscheduled maintenance, refueling, cargo upload and/or download, and passenger movement. The resources simulated comprise a runway, ramp parking spots, fuel resources, and cargo resources. Fuel trucks, truck refueling stands, fuel hydrant refueling pits, hydrant laterals, hydrant fuel tanks, and bulk fuel storage are the fuel related resources simulated in BRACE. The cargo resources include K-loaders, forklifts, loading docks, and warehouse storage facilities. Among the many possible performance measures that BRACE can provide for AMC decision makers are aircraft, cargo, or passenger throughput, or the number of resources required to meet a specific throughput [1].

MASS simulates the AMC global airlift system and is capable of simulating AMC policies, procedures, operations, aircraft, air bases, cargo, passengers, and support resources as they relate to the airlift system [11]. MASS simulates a fleet of aircraft moving a given amount of cargo and passengers from any number of on-load points, through any needed en-route stops, to any number of off-load points, then recovering and returning to home station for another mission. The model can continue this process for as many simulated days as desired, or until all requirements have been airlifted to their destination [11].

Both BRACE and MASS are large simulation models and significant amounts of time are required to accomplish any specific study using them. Reducing the time it takes to accomplish a BRACE or MASS study is of much interest to AMC. We demonstrate in this dissertation the use of an analytical model in concert with BRACE and MASS to reduce the time of studies based on the methods described above. In order to accomplish that goal the ACV bias problem is resolved and specific methods and justifications are developed for the surrogate search method. These issues are briefly discussed in the next section.

1.3 Dissertation Issues

1.3.1 ACV Bias Resolution under Known Probability Structures. Previous researchers have failed to provide a solution to the bias observed in the point estimates produced when employing the ACV method. To apply the ACV method to a real simulation model, this bias problem had to be resolved. A particular resolution to this problem for simulation models that possess specific properties has been completed and is presented in this dissertation. The research develops a general Monte Carlo method of distribution sampling that resolves the bias problem. The Monte Carlo method is demonstrated using a queueing network example. This Monte Carlo method is applicable if the mean and variance of the input random variables used to produce the ACV are known. Additionally the distribution of these random variables must be known either exactly or approximately. This is a simulation model specific requirement. In many cases, this requirement is not difficult to meet. The ACV Monte Carlo method is demonstrated to perform favorably when compared to internal and external CV's. Additionally the efficiency of the ACV Monte Carlo method is demonstrated.

1.3.2 ACV Bias Resolution without Complete Probability Knowledge. For some simulation models, the expected value of the moments of the input random variables—the inputs to the analytical model—are not always known. For example, a proportion that is required for the analytical model may be the result of a rule within the simulation model instead of a strict random number draw. In that case the mean and variance of the proportion are not known parameters of a random variable that is an input of the simulation model. Therefore, the Monte Carlo method described above cannot be used in those situations.

BRACE presents just such a situation when it simulates aircraft refueling. BRACE simulates aircraft refueling using both hydrant systems and fuel trucks. The mean time required to refuel an aircraft is different for the two types of refueling and some proportion of aircraft will be refueled by hydrant while the other aircraft are refueled by fuel truck. An appropriate analytical model that accounts for this difference in mean refueling time must also account for the proportion of aircraft receiving each type of refueling. In BRACE, this proportion is based on a simple rule: when aircraft arrive to the airfield they are parked at a ramp spot with a hydrant system refueling pit if one is available. Otherwise, it will be parked at a spot without a refueling pit and receive fuel from a fuel truck [1]. The proportion of aircraft that will be refueled at a fuel pit depends on many factors including aircraft arrival rates and the time aircraft spend on the airfield. This proportion can only be known as a result of performing simulation replications. MASS also provides numerous examples of analytical model inputs that have unknown distributional parameters.

Both non-parametric and parametric alternatives to the Monte Carlo method are presented in this dissertation using a simulation model based on BRACE. To conduct this research, a queueing network analytical model employing state-of-the-art techniques is developed to produce ACV's for the BRACE model. We apply non-parametric re-sampling methods such as the bootstrap [20] and SIMDAT [50] that do not require any knowledge of the moments or probability distributions of the analytical model inputs to estimate the ACV mean. The bootstrap method re-samples realized data points (vectors) while the SIMDAT method actually generates new *psuedo-data* points based on a non-parametric density function estimator. The re-sampled points from both methods are used to approximate the mean of the ACV when knowledge of the probability structure is unknown. A parametric approach is also used in this dissertation. As in the Monte Carlo method, the central limit theorem is invoked so that the distributions of the inputs are assumed to be normally distributed. The difference here is that not all parameters of the multivariate normal distribution are known. Different schemes of estimating all or some of those parameters are explored. Finally, combinations of these methods are also examined. The different methods are compared for their later use in this dissertation.

1.3.3 Surrogate Search. A new methodology is developed in order to justify and perform a surrogate search. The justification for using an analytical model in place of a simulation model is based on classic simulation model validation and verification techniques. The validation and verification techniques provide a framework for simulation analysts to demonstrate that a simulation model is a valid representation of the system under study. We adapt these methods for our own use in showing that the analytical models we employ are valid representations of the simulation model under study. The surrogate search validation and verification methods rely upon the results of the ACV method and are fully integrated into the steps necessary to complete a simulation study. Following our presentation on validating the analytical model as a surrogate, we derive and describe the methods necessary to perform a surrogate search.

To demonstrate the effectiveness of the surrogate search method, aspects of a response surface methodology (RSM) study are made on a simulation model based on BRACE and on the MASS simulation model itself. Simply put, RSM consists of several statistical techniques for empirical model building. The methodology describes a means of careful design and analysis of experiments in order to most efficiently relate a response (or output) random variable to the levels of a number of predictor (or input) random variables [12]. RSM can be applied to any number of systems including discrete event simulation models and is an excellent vehicle to develop the surrogate search idea since an RSM study can provide for many opportunities to conduct experimental searches within a single study. The use of an analytical model in conjunction with a simulation model during an RSM study can result in significant time savings since the number of simulation replications can be significantly reduced via variance reduction and the surrogate search methods

The RSM study conducted on the model based on BRACE consists of a simple two-factor design meant to illustrate the surrogate search method and highlight basic issues of the method. On the other hand, the RSM study conducted using MASS consists of a "real-world" sized problem using an actual Air Force simulation model. Several issues are addressed in this demonstration. First, an analytical model of MASS is constructed in order to complete the study, the first time such a model has been built. Secondly, two new performance measures for determining the most efficient movement of cargo are developed for the purposes of this study. Finally resolutions to several non-standard surrogate search issues are presented. In both studies, the surrogate search method is shown to achieve the goal of reducing simulation study times.

1.4 Overview

This dissertation is organized in the following manner. Following this introduction is a literature review chapter. The areas of control variates, mean value analysis queueing network models, and response surface methodology are each addressed. This is followed by a short chapter on the characteristics of simulation and analytical models. The next chapter presents the research completed on resolving the ACV method bias problem with known distribution parameters. A chapter follows this on resolving the bias problem without complete distributional knowledge. The surrogate search method is developed in the next chapter followed by a chapter that contains the two demonstrations of the method. The first demonstration is a simple RSM study conducted on a simulation model based on BRACE. The final demonstration is a real-world sized RSM study conducted using MASS. The final chapter is a discussion of contributions to the research community as a result of the research completed for this dissertation. Included will be recommendations for future research. A glossary of acronyms and abbreviations that appear in this document is included for the reader's benefit as an appendix.

II. Literature Review

2.1 Overview

The following discussion presents the pertinent literature on the three main topic areas required to complete this dissertation research—control variates (CV's), analytical modeling using the Mean Value Analysis (MVA) algorithm, and response surface methodology (RSM). The control variate section contains a presentation of the theory of control variates, control variate selection, and the main categories of control variates. The section on analytical modeling begins by describing the types of multi-class queueing networks that have exact analytical solutions followed by a description of MVA and the MVA algorithm. The section concludes with a discussion of an approximate MVA algorithm for networks with fork-join constructs. This type of analytical network is appropriate for one of the example studies under consideration. Finally, the last section in this chapter presents an overview of RSM.

2.2 Control Variates

The method of control variates is an effective and practical variance reduction technique for discrete event simulation. Advantages of this technique include low computational overhead and applicability to a wide range of models.

2.2.1 Control Variate Theory. Several textbooks present excellent references on the use of control variates. The Law and Kelton textbook [32] on simulation modeling is one example. These authors provide an overview on control variate theory, application, and a brief discussion on the difference between internal and external control variates. In addition to their excellent coverage of control variates, they also provide an extensive reference list of control variate literature.

Nelson [35] presents a guide for simulation practitioners for applying three variance reduction techniques, including control variates. Nelson provides methods for finding point and interval estimators, software requirements, and guidelines for experimental design when using control variates.

This article is designed as a tutorial, not as a definitive presentation of control variate theory. As such it is very useful for understanding how the theory of control variates is applied to actual simulation studies. The following discussion summarizes control variate theory as presented in Nelson [35] and Law and Kelton [32].

Suppose we are employing simulation to estimate $\mu = E[Y]$ where the random variable Y is the steady-state waiting time for customers in a queueing system. To reduce variance in the estimate of μ, we might exploit another random variable C (e.g. interarrival time), that is positively correlated with Y and has known expectation $\mu_C = E[C]$. A new random variable $Y(b) = Y - b(C - \mu_C)$ can be constructed for each simulation replication. The expectation

$$E[Y(b)] = E[Y] - b(E[C] - \mu_C) \tag{2.1}$$

is an unbiased estimator of μ for any real number b. Since

$$Var(Y(b)) = Var(Y) + b^2 Var(C) - 2bCov(Y, C) \tag{2.2}$$

it is clear that $Var(Y(b))$ will be lower than $Var(Y)$ if and only if

$$2bCov(Y, C) > b^2 Var(C) \tag{2.3}$$

This relationship holds if Y and C are highly correlated and b is selected appropriately. From Equation (2.2), the value of b that minimizes $Var(Y(b))$ is given by

$$\beta = \frac{Cov(Y, C)}{Var(C)} \tag{2.4}$$

Normally $Cov(Y, C)$ is not known but, for n simulation replications, β can be estimated by the moment estimator

$$\hat{\beta} = \frac{\sum_{j=1}^{n}(Y_j - \bar{Y})(C_j - \bar{C})}{\sum_{j=1}^{n}(C_j - \bar{C})^2} \tag{2.5}$$

where $\bar{Y}$ and $\bar{C}$ are the respective sample means of the n observations of Y and C. A controlled estimate of μ can then be obtained as

$$\bar{Y}(\hat{\beta}) = \bar{Y} - \hat{\beta}(\bar{C} - \mu_C) \tag{2.6}$$

We observe that, since $\hat{\beta}$ is not independent of Y and C, we cannot assume that $\bar{Y}(\hat{\beta})$ is in general unbiased.

This method can be generalized for $q > 1$ control variates $\mathbf{C} = (C_1, C_2, \ldots, C_q)'$ with respective known means $\boldsymbol{\mu}_{\mathbf{C}} = (\mu_1, \mu_2, \ldots, \mu_q)'$ to

$$\bar{Y}(\mathbf{b}) = \bar{Y} - \mathbf{b}'(\bar{\mathbf{C}} - \boldsymbol{\mu}_{\mathbf{C}}) \tag{2.7}$$

where each $\mathbf{b} = (b_1, b_2, \ldots, b_q)'$ is a vector of real numbers. If $\mathbf{b}$ is estimated in the same manner as in Equation (2.5) above, and Y and $\mathbf{C}$ are distributed multivariate normal, Lavenburg and Welch [31] have shown that

$$Var\left(\bar{Y}(\hat{\boldsymbol{\beta}})\right) = \frac{n-2}{n-q-2}(1 - R^2_{Y\ \mathbf{C}})Var(\bar{Y}) \tag{2.8}$$

where $\hat{\boldsymbol{\beta}} = (\hat{\beta}_1, \hat{\beta}_2, \ldots, \hat{\beta}_q)'$ is the vector of optimal estimates of $\mathbf{b}$, and $R^2_{Y\ \mathbf{C}}$ is the coefficient of multiple determination between Y and $\mathbf{C}$. This coefficient is computed as

$$R^2_{Y\ \mathbf{C}} = \frac{\boldsymbol{\sigma}_{Y\ \mathbf{C}}\, \Sigma_{\mathbf{C}}^{-1}\, \boldsymbol{\sigma}'_{Y\ \mathbf{C}}}{\sigma_Y^2} \tag{2.9}$$

where σ_Y^2 is the variance of Y, $\boldsymbol{\sigma}_{Y\,\mathbf{C}}$ and $\boldsymbol{\sigma}_{\,\mathbf{C}Y}$ are the covariance vectors between Y and $\mathbf{C}$, and $\Sigma_{\mathbf{C}}$ is the covariance matrix of $\mathbf{C}$. The value of $R^2_{Y\,\mathbf{C}}$ increases as each additional control variate is added, reducing variance. However, the term $(n-2)/(n-q-2)$ increases with the number of control variates. Depending on which effects are dominant, the addition of another control variate may cause the variance to grow rather than decrease [35]. Several methods have been proposed for effective control variate selection [2,9,39], that will be discussed later.

Lavenberg and Welch [31] offer a very detailed survey on the application of control variates. Other areas investigated include techniques for generating control variates and inefficiencies resulting from estimating control variate coefficients. Particularly useful are the two appendices of this article which provide detailed discussions on the application of control variates to generate confidence intervals. The first appendix describes the generation of confidence intervals using the method of independent replications (and equivalently the method of batch means) while the second appendix presents the generation of confidence intervals using the regenerative method of simulation.

Lavenberg and Welch [31] show that if Y and $\mathbf{C}$ have a joint multivariate normal distribution, the CV estimator is unbiased. Further, control variate application can be interpreted as a classical regression problem given the following joint normality assumption

$$\begin{bmatrix} Y \\ \mathbf{C} \end{bmatrix} \sim N_{1+q}\left[\begin{bmatrix} \mu_Y \\ \mu_{\mathbf{C}} \end{bmatrix}, \begin{bmatrix} \sigma_Y^2 & \sigma_{Y\,\mathbf{C}} \\ \sigma'_{\,\mathbf{C}Y} & \Sigma_{\mathbf{C}} \end{bmatrix}\right] \tag{2.10}$$

Computationally, this is a simple method for finding $\bar{Y}(\,\hat{\boldsymbol{\beta}})$ and estimating it's variance and associated confidence interval. Since

$$E[Y|\,\mathbf{C}=\,\mathbf{c}] = \mu + \,\boldsymbol{\beta}'(\,\mathbf{c} - \mu_c) \tag{2.11}$$

the regression problem can be stated as

$$\mathbf{Y} = \mathbf{X}\,\gamma + \varepsilon \tag{2.12}$$

where $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_n)'$, ε is the prediction error, and

$$\gamma = \begin{bmatrix} \mu \\ \boldsymbol{\beta} \end{bmatrix} \tag{2.13}$$

with

$$\mathbf{X} = \begin{bmatrix} 1 & c_{11} - \mu_1 & \cdots & c_{q1} - \mu_q \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ 1 & c_{1n} - \mu_1 & \cdots & c_{qn} - \mu_q \end{bmatrix} \tag{2.14}$$

where each c_{ij} is the ih control variate of the jh replication $(i = 1, 2, \ldots, q;\ j = 1, 2, \ldots, n)$. To find the least squares estimators set

$$\hat{\gamma} = \begin{bmatrix} \hat{\mu} \\ \hat{\boldsymbol{\beta}} \end{bmatrix} = (\,\mathbf{X}'\mathbf{X})^{-1}\,\mathbf{X}'\mathbf{Y} \tag{2.15}$$

resulting in the equation

$$\mathbf{Y} = \mathbf{X} \begin{bmatrix} \hat{\mu} \\ \hat{\boldsymbol{\beta}} \end{bmatrix} + \mathbf{e} \tag{2.16}$$

where $\mathbf{e}$ is the vector of least squares residuals. If we pre-multiply both sides of the equation by $n^{-1}\,\mathbf{1}'$, where $\mathbf{1}$ is a $1 \times n$ vector of 1's,

$$n^{-1}\,\mathbf{1}'\{\,\mathbf{Y}\} = n^{-1}\,\mathbf{1}'\left\{\,\mathbf{X}\begin{bmatrix}\hat{\mu}\\ \hat{\boldsymbol{\beta}}\end{bmatrix} + \mathbf{e}\right\} \tag{2.17}$$

that yields

$$\bar{Y} = \hat{\mu} + \hat{\boldsymbol{\beta}}'(\,\bar{\mathbf{C}} - \,\boldsymbol{\mu}_{\mathbf{C}}) \tag{2.18}$$

since the sum of the residuals is zero. It is clear by comparing equations 2.7 and 2.18 that $\hat{\mu}$ is equal to $\bar{Y}(\,\hat{\boldsymbol{\beta}})$. Therefore, by classical regression theory [37], the variance of $\bar{Y}(\,\hat{\boldsymbol{\beta}})$ is given by

$$Var(\bar{Y}(\,\hat{\boldsymbol{\beta}})) = s_{11}SE \tag{2.19}$$

where s_{11} is the upper left element of $(\,\mathbf{X}'\mathbf{X})^{-1}$ and MSE is computed as

$$\text{MSE} = \left(\frac{1}{n-q-1}\right)(\,\mathbf{Y}'\mathbf{Y} - \hat{\boldsymbol{\gamma}}'\mathbf{X}'\mathbf{Y}) \tag{2.20}$$

Then the $100(1-\alpha)\%$ confidence interval can be obtained by

$$\bar{Y}(\,\hat{\boldsymbol{\beta}}) \pm t_{1-\gamma/2,n-q-1}\sqrt{s_{11}SE} \tag{2.21}$$

Control variates can also be formulated for multivariate responses. Several papers present methods for using control variates for multi-response simulation models. See for example [9,38,39, 44,55,57]. The extension to multivariate response is straightforward with the univariate statistics replaced by their multivariate analogs.

Further, not all control variates are linear. Nelson [34] describes several different forms of control variate construction. Other possible control variance estimators include the *ratio CV*,

$$\bar{Y}_{ratio} = \left[\frac{\bar{Y}}{\bar{C}}\right] \mu_C \tag{2.22}$$

the *polynomial CV*,

$$\bar{Y}_{poly}(\beta) = \bar{Y} - \beta_1(\bar{C} - \mu_C) + \beta_2(\bar{C} - \mu_C)^2 + \dots \tag{2.23}$$

an extension of the ratio CV that includes a multiplier "β"

$$\bar{Y}_{mult}(\beta) = \bar{Y}\left[\frac{\mu_C}{\bar{C}}\right]^{1/\beta} \tag{2.24}$$

and the *power CV*

$$\bar{Y}_{power} = \bar{Y}^{\bar{C}/\mu_C} \tag{2.25}$$

2.2.2 Control Variate Bias. Lavenberg and Welch [31] proved that CV estimators are unbiased if Y and $\mathbf{C}$ have a joint multivariate normal distribution. However, if Y and $\mathbf{C}$ are not jointly normal, CV estimators are, in general, biased if $\boldsymbol{\beta}$ is estimated. Nelson [36] proved that regardless of the distribution of $(Y, \mathbf{C})$ the following central limit theorem for control variates holds

$$n^{1/2}\left[\bar{Y}(\hat{\boldsymbol{\beta}}) - \mu\right] \xrightarrow{D} N\left[0, \sigma_Y^2(1 - R_{Y\,\mathbf{C}}^2)\right] \text{ as } n \to \infty \tag{2.26}$$

Thus, even when the normality assumption is not appropriate, the asymptotic property 2.26 justifies the use of CV's as long as n is sufficiently large.

However, if n is not very large and the normality assumption doesn't hold, researchers have suggested several remedies. Tew and Wilson [51] present a method for checking the multivariate normality assumption. Nelson [36] presents several different methods to combat the bias problem including splitting, jackknifing, and bootstrapping. He makes several recommendations on the most appropriate method based on the number of replications. Avramidis and Wilson [3] describe a different splitting scheme that guarantees unbiased CV estimation without any distributional assumptions at the price of a "slight" increase in confidence interval width.

2.2.3 Control Variate Selection. Selection of the best, or nearly the best, subset of possible control variates is another area of extensive research. Nelson [35] describes two methods for deciding which subset of possible control variates to use for reducing variance. One method is to use a regression software package and perform stepwise regression on all possible control variates and then select the subset of controls that create the largest amount of variance reduction. He also proposes a means of determining a marginal improvement ratio for adding an additional control variate to the set of controls already in use. For example, for a set of q control variates, the marginal improvement ratio is computed by comparing $1 - R^2$ for the set of $q+1$ control variates, and $1 - R^2$ for q control variates, where R^2 is an estimate of the square of the multiple correlation coefficient of the response variable. He provides a table of marginal improvement ratios necessary for adding an additional control variate based on the number of replications performed and number of control variates already included.

Bauer and Wilson [9] develop a method for selecting the best subset of possible control variates for multi-response simulation models. Their criteria for selection minimizes the mean-square confidence region volume for responses under the assumption that $(\mathbf{Y}, \mathbf{C})$ is distributed multivariate normal. Porta Nova and Wilson [39] develop control variate selection criteria when estimating multi-response simulation metamodels, when $(\mathbf{Y}, \mathbf{C})$ is not jointly normal. They consider specific covariance structures for the responses and possible controls that occur for specific types of psychometric and econometric simulation studies.

2.2.4 Internal and External Control Variates. Existing control variate methods are generally classified as either *internal or external.* Internal control variates may be input random variables or arbitrary functions of these inputs, and deliberative selection of the right combination of candidate variables to produce a low-variance unbiased estimate of the response can be a complex undertaking. Alternatively, the external method requires the creation of an analytical model of a simplified version of the system under study. A simulation of the simplified system is then implemented using the same random number streams as the original simulation. External control variates are rarely used in practice, due to the difficulty in obtaining the necessary synchronization of the random variate generators [32].

Examples of external control variates can be found in Burt, Gaver, and Perlas [14] and Gaver and Shedler [23]. In [14], the effectiveness of several variance reduction techniques on project graph analysis (PERT, GERT, CPM, etc.) network simulations is examined. Among the different techniques are external control variates. The authors generate the external control variates by first constructing simplified networks that have analytical solutions and are "similar" to the networks under study. They then simulate both models separately using a common random number stream to drive both simulations. Gaver and Shedler [23] apply external control variates to simulations of a multiprogrammed computer system. In a similar manner as in the previous article, they propose a model of the system and a similar model that can be solved analytically. In both articles significant reduction in variance is reported.

Many researchers have proposed many forms of internal control variates. We will focus on those CV's that will be considered in this dissertation. Standardized work variables are developed for queueing systems with the regenerative property in Wilson and Pritsker [56]. Given a service process $U_j(k) : j \geq 1$, at service center k with known expected value μ_k and variance σ_k^2, standardized work variables are defined as

$$C_k(t) = [a(k,t)]^{-1/2} \sum_{j=1}^{a(k,t)} [U_j(k) - \mu_k]/\sigma_k \tag{2.27}$$

where $a(k,t)$ is the number of service times started at center k during the time period $[0,t]$. These standardized variables were developed since previous internal control variates [30] have been shown to have an asymptotic variance equal to zero. This property causes the variance covariance matrix for a set of these controls to be asymptotically singular. Since an inverse of this matrix (or usually an estimate of it) must be computed to find (or estimate) the optimal control coefficient, numerical problems can arise as replication length increases. The standardized work variables, on the other hand, have an asymptotic variance of 1. Further, Wilson and Pritsker prove that standardized control variables for queues with the regenerative property converge in distribution to multivariate normal with a mean vector of $\mathbf{0}$. Experiments on a simple network were performed with substantial variance reduction reported.

Another set of standardized internal control variables, standardized routing variables, are developed in Bauer and Wilson [10]. These can be used for discrete-event simulation models that have a multinomial construct. Standardized routing variables attempt to exploit the correlation between departures from the mean branching probabilities in a network and the resultant network response. They are defined in the following manner. Consider a multinomial branching process of g branches leading to g service centers, and define an indicator variable as

$$I_i(j) = \begin{cases} 1 \text{ if the } i-th \text{ departing customer goes to center } j, \\ 0 \text{ otherwise} \end{cases} \tag{2.28}$$

Then a standardized routing variable for center j is defined as

$$R_j = \sum_{i=1}^{N(t)} \frac{I_i(j) - p(j)}{\{N(t)[1 - p(j)]p(j)\}^{1/2}}, \quad j = 1, 2, \ldots, g \tag{2.29}$$

where N(t) is the total number of transits through all g branches in the time interval $[0, t]$ and $p(j)$ is the probability that a customer is sent to center j. Bauer and Wilson performed several experiments on a simple network with results indicating significant confidence interval length reduction, particularly when the standardized routing variables are used in conjunction with standardized work variables.

2.2.5 Analytical Control Variates. For many simulation studies, it may be possible to avoid the technical challenges of external or multiple internal control variates by using an external, analytical model to generate a single "analytical" control variate (ACV) for each replication of a simulation. Nelson [34] first presented the idea of an analytic control variate, but did not report any experimental results. Sharon [48] and Sharon and Nelson [49] used Jackson network results to generate analytical control variates for queueing network simulations. Tomick [53] and Tomick, Litko, and Bauer [54] demonstrated that the approach can produce significant variance reduction in a broader range of queueing network models. All of these researchers also reported unacceptable levels of bias in the response estimate.

Sharon and Nelson [49] describe the construction of an ACV by first describing the construction of an external control variate (ECV). As described above, ECVs require a second system that is similar, yet different from the simulation system of interest. In fact, the second system has a known analytical or numerical approximation of ϕ, where ϕ is an output performance measure of the second system that corresponds to θ, the output performance measure of interest of the primary simulation system. Calling the second system the *control system*, both systems are simulated using common random numbers. If C, the estimator of ϕ from the control system, is strongly correlated to Y, the estimator of θ for the system of interest, variance reduction will occur. Since ECVs require a second simulation, significant levels of variance reduction must occur for ECVs to be an efficient means of variance reduction.

Nelson [35] suggested another method of generating an external control variate. In this case, let δ be a vector of parameters for the input distributions for the control system such that $\phi = g(\delta)$, where g is a function and ϕ is the same as described above. If the system of interest has the same input parameters, δ, but is different from the control system in other ways, the new method can be applied. The authors point out that the key step is to simulate the system of interest to obtain Y and $\hat{\delta}$, where $\hat{\delta}$ is an estimator of the known quantity δ. The *analytical control variate* is then formed by $Y(b) = Y - b\left(g(\hat{\delta}) - g(\delta)\right)$. For K replications $(Y_k, \hat{\delta}_k), k = 1, 2, \ldots, K$ an ACV controlled estimator is

$$\bar{Y}(\hat{b}^*) = \bar{Y} - \hat{b}^*\left(\bar{g}(\hat{\delta}) - \phi\right) \tag{2.30}$$

where $\hat{b}^*$ estimates $b^* = Cov\left[Y, g(\hat{\delta})\right] / Var\left[g(\hat{\delta})\right]$. The authors point out that $g(\hat{\delta})$ may not be an unbiased estimator of ϕ.

One advantage of an ACV to an ECV, according to the authors, is that an ACV doesn't require the extra time required to generate a second simulation. Also, the problems associated with applying common random numbers are not encountered. All that is required is to apply ACVs are a control system, the function g (which may be a numerical approximation or closed form function), and the vector of realized input parameters $\hat{\delta}$.

The authors perform experiments on simple queueing networks to explore the effects of an ACV. Their results indicate significant variance reduction under most of the experimental settings. However, they do experience bias in results. They point out that bias of the controlled estimator can arise from the estimation of b^* and since $E\left[g(\hat{\delta})\right] \neq g(\delta)$. The authors point out that these are important effects that must be investigated since variance reduction at the expense of increased bias may not be acceptable [49].

2.3 Analytical Modeling

When constructing an analytical model for the purpose of generating an ACV, the model need only be a reasonably accurate representation of the modeled system. Essentially, the model should be a function that uses some subset of the same input random variables, or their moments, as the simulation model and has as its output the same measures of interest. Additionally, the analytical model should be easy to implement and reasonably fast. To meet these requirements, the ACV analytical model considered for this dissertation will be a closed multi-chain queueing network. By using a Mean Value Analysis (MVA) algorithm, the steady state expected values of the measures of interest can be found rapidly and exactly [13, 16, 29]. Such an approach is demonstrated in this dissertation to achieve significant levels of variance reduction at a low computing cost when applied to a small simulation model of a multi-programmed computer system. The following sections present discussions on product form networks, the MVA algorithm, and a fork-join queueing network model.

2.3.1 Product Form Networks. The basic type of product form queuing networks are known as BCMP networks, named for Baskett, Chandy, Muntz, and Palacios. These authors describe a general multi-chain queueing network and demonstrate that it has a product form solution [8]. At the time of their publication, BCMP networks described the most general network with a known product form solution. Although several authors have since extended their work in very specific areas (see [26] for example), BCMP networks still form the basic guidelines for product form networks. For that reason, we will first concern ourselves with networks that meet the BCMP requirements.

BCMP networks are open, closed, or mixed networks of queues or *service stations* with one or more classes of customers that have one of the four following service disciplines:

First Come First Served (FCFS) Customers are served in the order of their arrival by a single server. The service time for all customers is exponential with the same mean for all classes of customers. Load dependent service rates are allowed.

Processor Sharing (PS) Customers are serviced by a single server with a processor sharing (time division) service discipline. The service time distribution can be any distinct Coxian distribution for each class of customer. Load dependent service rates are allowed.

Delay Station (D) This service station has an infinite number of servers (or at least as many as the total number of customers in the network). Each class of customer may have a distinct Coxian service time distribution.

Last Come First Served (LCFS) There is a single server and the service discipline is last come first served where the last arriving customer has an absolute priority (the currently serviced customer is immediately pre-empted). Each class of customer may have a distinct Coxian service time distribution with load dependent service rates.

Each class of customer in the network may have its own probabilistic routing chain and multiple servers can be represented by using load dependent service rates. Based on these assumptions, Basket, et. al. [8] provide product form formulas for the equilibrium state probabilities. They prove that the formulas are correct by showing that they satisfy the independent balance equations. Once the state probabilities are found, the different network performance measures, such as mean queue size, mean waiting times, and throughput, can be determined.

2.3.2 Mean Value Analysis. Mean value analysis can provide the same performance measures for closed or capacitated BCMP networks without the need to solve for the equilibrium state probabilities [29,42]. In particular, MVA can calculate mean response time R_i (waiting and service time at service center i), throughput λ_i, queue length Q_i (number of customers waiting and in service at station i), and server utilization U_i (expected number of busy servers at station i). Detailed explanations of the MVA algorithm can be found in Bruell and Balboa [13] and in Conway and Georganas [16].

The foundation of MVA is the *arrival theorem* first proven by Lavenberg and Reiser [29]. The theorem states that for a closed network with N customers, an arriving customer to service center i observes the same distribution of customers at that station as the stationary (random observer's) distribution for the same network with $N-1$ customers. Consider a single class network with m service stations where the service rate at station i when n customers are present is given by $\mu_i(n)$. Then the arrival theorem can be used to develop the *marginal local balance theorem*:

$$\mu_i(n)P_i(n|N) = \lambda_i(N)P_i(n-1|N-1) \tag{2.31}$$

where $P_i(n|N)$ is the probability that n customers are at station i given that N customers are in the network, and $\lambda_i(N)$ is the customer throughput at station i when N customers are in the network. By applying the marginal balance equation recursively, the performance measures can be computed. Mean queue length can be found by

$$Q_i(N) = \sum_{n=1}^{N} nP_i(n|N) = \sum_{n=1}^{N} \frac{n\lambda_i(N)}{\mu_i(n)} P_i(n-1|N-1) \tag{2.32}$$

The value of throughput $\lambda_i(N)$ in Equation (2.32) is unknown, but by applying Little's Law we find that

$$R_i(N) = \sum_{n=1}^{N} \frac{n}{\mu_i(n)} P_i(n-1|N-1) \tag{2.33}$$

If station i has only one server, then

$$R_i(N) = s_i \sum_{n=1}^{N} nP_i(n-1|N-1) = s_i[1 + Q_i(N-1)] \tag{2.34}$$

where s_i is the mean service rate of a single server at station i. If station i is an infinite server station, then $R_i(N) = s_i$ for all N.

Both Equations (2.33) and (2.34) relate the response time of any station when N customers are in the system to the distribution of customers at the same station when $N-1$ customers are in the network. Thus, by beginning calculations when $N = 1$ (so that $P_i(0|N-1) = 1$ and $Q_i(N-1) = 0 \ \forall i$), all performance measures can be found recursively. To find the station throughputs for each iteration, the average cycle time for a customer at an arbitrary reference station must be calculated. If we use station 1 as the reference station, the average time between departures from the reference station for the same customer is given by

$$CT_1(N) = \sum_{i=1}^{M} \frac{v_i R_i(N)}{v_1} \tag{2.35}$$

where each ratio v_i/v_1 is the mean number of visits a customer makes to station i for every visit to station 1. Then station throughput can be solved for by

$$\lambda_i(N) = \frac{N v_i}{CT_1(N) v_1} \tag{2.36}$$

Using Little's Law, the mean queue length and station utilization can be found by

$$Q_i(N) = R_i(N)\lambda_i(N) \tag{2.37}$$

$$U_i(N) = s_i \lambda_i(N) \tag{2.38}$$

If station i is a single server station, the results from Equation (2.37) can be applied to Equation (2.34) to calculate the mean response times with one more customer in the system. However, if station i has more than 1 server, the marginal local balance theorem can be used to determine the new distribution of customers as

$$P_i(n|N) = \frac{\lambda_i(n)P_i(n-1|N-1)}{\mu_i(n)}, \qquad n > 0 \tag{2.39}$$

$$P_i(0|N) = 1 - \sum_{n=1}^{N} P_i(n|N) \tag{2.40}$$

These probabilities can be applied to Equation (2.33) to calculate the multiple server response times for the next iteration. This process is repeated until N equals the total number of customers in the network. These formulas can be extended to account for multi-class networks. Appropriate algorithms can be found in Bruell and Balboa [13] and Conway and Georganas [16].

2.3.3 Fork-Join Queueing Network Approximation. One important construct of BRACE that should be included in an analytical model for ACV use is that of concurrent service activities. These types of activities cannot be modeled as a product form network, so the MVA algorithm cannot be directly applied [29]. However, a fork-join approximation method for MVA has been developed by Dietz and Jenkins [19] from the work of Rao and Suri [41]. The fork-join approximation allows for the modeling of concurrent activities within the network. A fork-join node can be described in the following manner. A customer arriving at a fork-join node generates clones that enter the separate substations of the node and are rejoined to the parent customer once service is completed at their separate substations. Once all clones complete their servicing, the parent customer is made whole and can move on to the next network service station according to its routing chain.

Dietz and Jenkins [19] derive a method within the MVA framework to approximate network measures for multiple fork-join nodes with multiple server activities and probabilistic service requirements. Their method can be summarized in the following manner. Consider a fork-join node i that contains K_i substations where the probability that a clone proceeds to substation k of station i is $q_{ik} = 1$ for all $k = 1, \ldots, K_i$. Approximate mean response time, queue length, and utilization for clones for substations ik are represented by $R_{ik}(N)$, $Q_{ik}(N)$, *nd* $U_{ik}(N)$. Two approximations are made in order to evaluate network performance measures:

- *Approximation 1.* For a network with N customers, a clone arriving at a substation sees the stationary distribution of clones at the substation for the same network with $N-1$ customers.
- *Approximation 2.* The response time experienced by a clone at a substation can be represented as a exponentially distributed random variable and is independent of the response time at other substations.

Using Approximation 2, let the response time for any substation ik be denoted by the exponential random variable $T_{ik}(N)$ with rate parameter $\theta_{ik}(N) = 1/R_{ik}(N)$. This response time represents both substation service time and any waiting time in the substation. The mean time that a parent customer spends at fork-join node i is given by $E[\max_{k=1,\ldots,K_i} \{T_{ik}(N)\}]$. This time is then used to calculate network cycle time.

If we let S represent the subset of all possible substations that any particular customer will require at fork-join node i, then the assumption that $q_{ik} = 1$ for all $k = 1, \ldots, K_i$ can be relaxed by conditioning on S. Let Ω_i be the union of all possible subsets for a particular fork-join node i, and let $\pi_i(S)$ be the probability that subset S is required by a customer. Note that the number of subsets in Ω_i is given by $\sum_{k=0}^{K_i} \binom{K_i}{k} = 2^{K_i}$. Assuming that the selection of substations is independent, the probability that a customer requires any particular subset S at fork-join node i

is given by

$$\pi_i(S) = \prod_{k \in S} q_{ik} \prod_{k \notin S} (1 - q_{ik}) \tag{2.41}$$

The mean of the conditional holding time at fork-join node i is $E[\max_{k \in S} \{T_{ik}(N)\}]$ (defined as zero if $S = \emptyset$).

Finding the conditional mean holding time is mathematically the same as determining the mean time to failure for a parallel system of independent components with exponentially distributed failure times [25]. By the independence described in Approximation 2, the cumulative distribution function (CDF) for conditional holding time is

$$\begin{aligned} F(t) &= \prod_{k \in S} P\{T_{ik}(N) \leq t\}, \\ &= \prod_{k \in S} (1 - \exp\{-\theta_{ik}(N)t\}) \end{aligned} \tag{2.42}$$

Since $E[X] = \int_0^\infty (1 - F(t))dt$ for any nonnegative continuous random variable X with CDF $F(t)$, then

$$\begin{aligned} E[\max_{k \in S} \{T_{ik}(N)\}] &= \int_0^\infty \left\{ 1 - \prod_{k \in S} (1 - \exp\{-\theta_{ik}(N)t\}) \right\} dt \\ &= \int_0^\infty \left\{ 1 - 1 + \sum_{k \in S} \exp\{-\theta_{ik}(N)t\} \right. \\ &\quad - \sum_{k \in S} \sum_{\substack{l \in S \\ l < k}} \exp\{-(\theta_{ik}(N) + \theta_{il}(N))\, t\} \\ &\quad + \sum_{k \in S} \sum_{\substack{l \in S \\ l < k}} \sum_{\substack{m \in S \\ m < k,l}} \exp\{-(\theta_{ik}(N) + \theta_{il}(N) + \theta_{im}(N))\, t\} \\ &\quad \left. - \cdots + (-1)^{K(S)+1} \exp\left\{ - \left(\sum_{k \in S} \theta_{ik}(N) \right) t \right\} \right\} dt \end{aligned} \tag{2.43}$$

where $K(S)$ is the number of substations in S. Evaluating the integral yields

$$\begin{aligned} E[\max_{k\in S}\{T_{ik}(N)\}] &= \sum_{k\in S}\frac{1}{\theta_{ik}(N)} - \sum_{k\in S}\sum_{\substack{l\in S\\ l<k}}\frac{1}{\theta_{ik}(N)+\theta_{il}(N)} \\ &+\sum_{k\in S}\sum_{\substack{l\in S\\ l<k}}\sum_{\substack{m\in S\\ m<k,l}}\frac{1}{\theta_{ik}(N)+\theta_{il}(N)+\theta_{im}(N)} \\ &-\cdots+(-1)^{K(S)+1}\frac{1}{\sum_{k\in S}\theta_{ik}(N)} \end{aligned} \tag{2.44}$$

The MVA algorithm can be adjusted for a network with a set I of simple service stations and a set J of fork-join nodes. Response times for simple service stations are computed at each iteration using Equations (2.33) and (2.34). For the fork-join substations with multiple servers, Approximation 1 allows for application of the marginal local balance theorem and Little's Law so that

$$R_{ik}(N) = \sum_{n=1}^{N}\frac{n}{\mu_{ik}(n)}P_{ik}(n-1|N-1) \tag{2.45}$$

For single-server fork-join substations, Approximation 1 leads to

$$R_{ik}(N) = s_{ik}\left\{1+Q_{ik}(N-1)\right\} \tag{2.46}$$

Then the substation response rates $\theta_{ik}(N) = 1/R_{ik}(N)$ can be used to find the mean conditional holding times for the fork-join nodes. Then cycle time for station 1 is given by

$$CT_1(N) = \sum_{i\in I}\frac{v_iR_i(N)}{v_1} + \sum_{i\in J}\frac{v_i}{v_1}\sum_{S\subseteq\Omega_i}\pi_i(S)E[\max_{k\in S}\{T_{ik}(N)]\} \tag{2.47}$$

Now that cycle time has been computed, Equations (2.36) - (2.38) can be used as before to compute throughputs, queue lengths, and utilization for the simple service stations. Since each fork-join substation ik is visited by a clone an average of q_{ik} times every time a customer visits node i, the substation throughput is calculated by

$$\lambda_{ik}(N) = \frac{N v_i q_{ik}}{CT_1(N) v_1} \tag{2.48}$$

The other substation performance measures are found using Little's Law by

$$Q_{ik}(N) = R_{ik}(N)\lambda_{ik}(N) \tag{2.49}$$

$$U_{ik}(N) = s_{ik}\lambda_{ik}(N) \tag{2.50}$$

An iteration of the algorithm can be started with the use of Equation (2.33) or (2.34) for a simple service station or Equation (2.46) for fork-join nodes with single servers. For fork-join nodes with multiple server substations, Approximation 1 provides clone probability distributions

$$P_{ik}(n|N) = \frac{\lambda_{ik}(n) P_{ik}(n-1|N-1)}{\mu_{ik}(n)}, \qquad n > 0 \tag{2.51}$$

$$P_{ik}(0|N) = 1 - \sum_{n-1}^{N} P_{ik}(n|N) \tag{2.52}$$

These probabilities can then be used in Equation (2.45) to calculate new response times. Iterations are continued until N is the desired number of customers in the network.

Dietz and Jenkins [19] tested their fork-join approximation on an aircraft sortie generation model. Their method provided highly accurate estimates of the mean performance measures for the networks studied in just a few seconds. The results were most accurate when resource utilization was relatively low.

Dietz extended the fork-join heuristic approach to include the possibility of multiple stations on a fork-join path, including nested fork-join constructs [17]. This is accomplished by changing approximation 2 to read:

- *Approximation 2.* The transit time of a clone along a fork-join path can be represented as an exponentially distributed random variable and is independent of the transit time for clones on other paths.

As before, the response time for any fork-join substation is solved as before. For fork-join paths with more than one substation, or a nested fork-join path, the transit time along the path is the sum of all sub-station response times and fork-join transit times on that path, where each fork-join path transit time is represented by an exponentially distribute random variable [17]. Dietz demonstrates the accuracy of the MVA heuristic by comparing results of the heuristic to that of a simulation model of military airlift field. His results indicate no relative error exceeding 13%, with most errors being much smaller for the parameters investigated.

2.4 Response Surface Methodology

Response surface methodology (RSM) is a set of statistical and mathematical techniques for empirical model building. The underlying mechanisms of some phenomena, or system, are understood well enough that mathematical models that are the result of this understanding can be derived. Analytical models of stochastic systems are one example. RSM is concerned with systems that are not understood well enough to allow for this approach, hence the term empirical model building. RSM encompasses:

1. Designing a series of experiments that will yield adequate and reliable measurements of the response(s) of interest in a region of interest.
2. Analyzing the results of those experiments to determine a mathematical model that best fits the data collected.

3. Searching for the optimal settings of the input variables. [27]

Discrete event simulation models are natural candidates for an RSM study. Simulation models are created because the mathematical relationships that govern the behavior of a system are either not understood or are intractable. Further, the output of a simulation model is a random variable. Since RSM studies require simulation replications at numerous experimental design points and gradient search points, the studies can require a significant amount of time to conduct. Exactly for those same reasons, the ACV and surrogate search could reduce that time appreciably. Later in this dissertation, these time saving methods are demonstrated on an example RSM study. In order to apply the RSM techniques, a brief primer on RSM will be presented here. The primer is primarily adapted from *Empirical Model-Building and Response Surfaces* by Box and Draper [12]. The areas discussed include empirical models, least squares analysis, design of experiment, steepest ascent, second-order model fitting, and exploration of maxima and ridge systems.

2.4.1 Empirical Models. In the introduction, some of the different mathematical models that describe systems were discussed. RSM outlines a means of deriving a statistical model, or empirical model, of a system. The assumption is that there exists some unknown functional relationship

$$E(y) = f\left(\xi_1, \xi_2, \cdots, \xi_k\right) \tag{2.53}$$

between the expected response of the system under investigation, y, and some number, k, of quantitative predictor variables, $\xi_1, \xi_2, \cdots, \xi_k$. Since this relationship is unknown, it is approximated by a polynomial approximation, often called a *graduating function*, over a specified region of interest [12]. In deriving the graduating function, it is convenient to convert the input variables to *coded* or *standardized* variables. If the current region of interest is defined for ξ_i is $\xi_{i0} \pm S_i$ where

ξ_{i0} is the center of the region, the coded variable x_i is defined by

$$x_i = \frac{\xi_i - S_i}{S_i} \tag{2.54}$$

The polynomial graduating function of the coded input variables, $x_1, x_2, \ldots, x_k$ is a linear combination of powers and products of the x's. A polynomial term is of order j if it contains the product of j of the x's, where some or all may be repeated. In other words, the terms $x_1^4, x_1 x_2^2 x_3, x_2^3 x_4$ are all of order 4. A polynomial is of order, or degree, d if the highest order term in the polynomial is of order d. The general form of the polynomial graduating function for $k = 2$ is given by

$$\begin{aligned} g(\mathbf{x}, \boldsymbol{\beta}) &= \beta_0 + (\beta_1 x_1 + \beta_2 x_2) + (\beta_{11} x_1^2 + \beta_{22} x_1 x_2 + \beta_{12} x_2^2) \\ &\quad + (\beta_{111} x_1^3 + \beta_{222} x_2^3 + \beta_{112} x_1^2 x_2 + \beta_{122} x_1 x_2^2) + \ldots \end{aligned} \tag{2.55}$$

where the β's are coefficients or (empirical) *parameters* that are estimated from the data.

Box and Draper [12] point out that the polynomial graduating function of degree d can be considered a Taylor's series expansion of the true underlying function $f(\boldsymbol{\xi})$ truncated after terms of degree d. They state that the following will usually be true:

1. The higher the degree of the approximating function, the closer the Taylor series can approximate the true function.

2. The smaller the region over which the approximation is made, the better the approximation for a polynomial function of a given order.

RSM is normally concerned with only first-order and second-order graduating functions. For $k = 2$ predictor variables the first-order graduating function is given by

$$g(\mathbf{x}, \boldsymbol{\beta}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \tag{2.56}$$

and the second-order graduating function is given by

$$g(\mathbf{x}, \boldsymbol{\beta}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2 \tag{2.57}$$

2.4.2 Least Squares Analysis. Least squares, or regression, analysis is a well-known statistical method of fitting empirical functions to data. Least squares analysis is used to fit the first and second-order graduating functions necessary to perform a RSM study. At this point in the discussion, only first-order graduating functions are considered since that is the beginning point of almost all RSM studies [12]. Since the theory is so well known only selected highlights of least squares analysis and how they apply to RSM will be presented here. Box and Draper [12] provide an excellent section on least squares analysis. For further study, Neter [37] is a thorough reference on the theory and application of least squares analysis.

Box and Draper point out that in performing RSM, an analyst is attempting to elucidate some model

$$y = f(\boldsymbol{\xi}, \boldsymbol{\theta}) + \epsilon \tag{2.58}$$

where

$$E(y) = f(\boldsymbol{\xi}, \boldsymbol{\theta}) \tag{2.59}$$

is the expected level of the response y given the k predictor variables $(\xi_1, \xi_2, \dots, \xi_k) = \boldsymbol{\xi}'$. In addition, there are p parameters $(\theta_1, \theta_2, \dots, \theta_p) = \boldsymbol{\theta}'$ and ϵ is the experimental error. To investigate this model, the analyst performs a series of experiments at n different settings $\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \dots, \boldsymbol{\xi}_n$ observing the corresponding $y_1, y_2, \dots, y_n$ values of the response. There are two important questions about the resulting function that must be answered:

1. Does the suggested model adequately represent the data?
2. Assuming the model does adequately model the data, what are the best estimates of the model parameters?

In practice, the second question is addressed first using the method of least squares to estimate the model parameters. The method of least squares selects the best estimates of $\boldsymbol{\theta}$ that minimizes the sum of squares of the errors given by

$$S(\boldsymbol{\theta}) = \sum_{u=1}^{n} [y_u - f(\xi_u, \boldsymbol{\theta})]^2 \tag{2.60}$$

$S(\boldsymbol{\theta})$ is referred to as the sum of squares function. For any given choice of p parameters of $\boldsymbol{\theta}$, there is a specific value of $S(\boldsymbol{\theta})$. The minimizing choice of $\boldsymbol{\theta}$ is the least squares estimate and is denoted by $\hat{\boldsymbol{\theta}}$. As long as the experimental errors $\epsilon_u = y_u - f\left(\boldsymbol{\xi}_u, \hat{\boldsymbol{\theta}}_u\right)$ are statistically independent, with constant variance and normally distributed, the least squares estimate is a "good" estimate [12]. Both Box and Draper [12] and Neter [37] provide the formulas necessary to find the least squares estimate of the empirical model.

The first question, "does the suggested model adequately represent the data", is answered in many different ways. Essentially, calculations and statistical tests can be performed to assess model adequacy. The primary guides to model adequacy are the size of the mean square error (MSE), the coefficient of multiple determination (R^2), and the F test for regression relation.

MSE is defined in the following manner. From least squares analysis, the relationship between the vector of observed responses, $(y_1, y_2, \cdots, y_n) = \mathbf{y}'$, the vector of fitted responses $(\hat{y}_1, \hat{y}_2, \cdots, \hat{y}_n) = \hat{\mathbf{y}}'$ and the vector of residuals $(e_1, e_2, \cdots, e_n) = \mathbf{e}'$ is given by

$$\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}} \tag{2.61}$$

It turns out that the vectors $\hat{\mathbf{y}}$ and $\mathbf{e}$ are orthogonal to each other [12], so that $\mathbf{e}'\,\hat{\mathbf{y}} = \hat{\mathbf{y}}'\,\mathbf{e} = 0$. This results in the relation

$$\mathbf{y}'\,\mathbf{y} = \hat{\mathbf{y}}'\,\hat{\mathbf{y}} + \mathbf{e}'\,\mathbf{e} \tag{2.62}$$

so that the sum of square errors (SSE) is given by

$$\mathrm{SSE} = \mathbf{e}'\,\mathbf{e} = \mathbf{y}'\,\mathbf{y} - \hat{\mathbf{y}}'\,\hat{\mathbf{y}} \tag{2.63}$$

Then MSE is given by SSE$/(n-p)$ where n is the number of observations and p is the number of parameters estimated. MSE is useful in assessing model adequacy since it is a measure of the variability within the residuals and, thus, provides a measure of how well the fitted responses match those actually observed [37]. Therefore, the smaller the value of MSE, the better the model.

The next measure for the adequacy of the fitted model discussed is the coefficient of multiple determination, R^2 given by

$$R^2 = \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \tag{2.64}$$

It can be shown that $0 \leq R^2 \leq 1$ and that R^2 represents the proportion of the variability within the observed responses that can be explained or accounted for by the model [12]. To understand this, first consider that the average of the observed responses is equal to the average of the fitted responses [37]. Using this fact, express R^2 by

$$R^2 = \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2 / (n-1)}{\sum_{i=1}^{n} (y_i - \bar{y})^2 / n - 1} = \frac{S_{\hat{y}}^2}{S_y^2} \qquad (2.65)$$

so that it is evident that R^2 is the ratio of the sample variance of the fitted values (a measure of the variability within the responses as explained by the model) to the sample variance of the observed values (a measure of the total variability within the observed responses). Hence the closer the value of R^2 to 1, the more the analyst can assume the model adequately fits the data.

Recalling the assumption that experimental error terms are independently and identically distributed normal random variates, a statistical test for model adequacy can be made. It turns out that if this assumption holds, the ratio of the mean square regression, $(\hat{\mathbf{y}}' \, \hat{\mathbf{y}} - n\bar{y}^2)/(p-1)$, to the mean square error, $\mathbf{e}' \, \mathbf{e}/(n-p)$ has an F distribution with $p-1$ and $n-p$ degrees of freedom, where n is the number of observations and p is the number of parameters estimated. The statistical test is posed in the following manner:

$$\begin{aligned} H_0 : \boldsymbol{\theta} &= \mathbf{0} \ (\theta_1 = \theta_2 = \cdots = 0) \\ H_1 : \boldsymbol{\theta} &\neq \mathbf{0} \ (\text{at least one } \theta_i \neq 0; \ i = 1, 2, \ldots, p) \end{aligned} \qquad (2.66)$$

Then the test statistic, $F = \text{SSR/MSE}$ is compared to the critical value $F_{\alpha,p-1,n-p}$. If the test statistic exceeds the critical value, the null hypothesis is rejected and the conclusion is that the fitted model is significant.

In summary, least squares analysis provides for the best means of estimating the parameters of the graduating function and the size of MSE, R^2, and the F test are the three primary measures of model adequacy. Therefore, a "good" model will:

- be significant, as indicated by a "large" value of the F test statistic;
- have a "small" MSE; and
- have a large R^2.

Other statistical tests exist for testing the significance of each of the estimated parameters and lack of fit for further model testing and perfecting. Each is described in Box and Draper [12] and Neter [37].

2.4.3 Design of Experiment. The graduating function relates the response and predictor variables of the system under study. Least squares analysis details the method of constructing the graduating function given some experimental data. The logical next step is to design the experiments that are to be conducted in order to collect the data. Many decisions must be made when designing the experiments. These include the number of experiments and the levels of the predictor variables for each of the experiments. Since each experiment involves the use of valuable resources (money, equipment, raw materials, labor, and time are some examples) one of the primary goals of any experimental design is efficiency. In other words, the analyst wishes to perform the least number of experiments that will result in an empirical model of sufficient accuracy. In addition, the analyst will normally wish to interact freely with the data, to make comparisons, find similarities, and to identify trends. One class of experimental designs that meets these requirements is the *factorial* experimental designs. They possess the following properties [12]:

1. They allow for numerous comparisons and so facilitate model creation and criticism.
2. They produce estimates of the parameters whose variance is as small, or nearly so, as those produced by any design covering the same region, hence they are highly efficient.
3. The parameter estimates are easy to calculate.

Factorial designs will be discussed in some detail below followed by a short discussion on two other types of experimental design, blocking and fractionating, that are based on factorial designs. Finally, some of the other possible exploratory designs will be mentioned.

One factorial design that is especially useful at the exploratory stage of a study, when not much is known about the system, is the *two-level* factorial design [12]. They are also useful as a first building block for developing many other experimental designs. Two-level factorial designs are designated by 2^k, where k is the number of predictor variables and 2^k is the number of design points. Often the predictor variables that are changed during the course of experimentation are referred to as factors. If we consider a design point as the particular combination of levels for all the factors, the factorial design consists of all 2^k design points where each predictor variable level is set to one of two levels. Using the coded predictor variables, this can be represented by

$$(x_1, x_2, \ldots, x_k) = (\pm 1, \pm 1, \ldots, \pm 1) \tag{2.67}$$

where every possible combination of ± signs is selected in turn. The design can be thought of geometrically where each design point is one of the vertices in a k dimensional hypercube. The designs are often listed in what is called *standard order* when designing the experiments or listing the results. One way of doing this is listing -1 and +1 alternatively in the x_1 column for a total of 2^k times. Under the x_2 column list alternate -1 -1 and +1 +1 pairs, under the x_3 column alternate fours of -1 -1 -1 -1 and +1 +1 +1 +1 and so on. A standard order listing for a 2^3 factorial design is shown in Table 2.1.

The main effect for a given predictor variable is defined as the average difference in the response level as the level of the predictor variable is changed from low to high. Factorial designs also allow for the calculation of interaction effects as well as that of the main effects. Interaction between factors occurs when the difference in response between the levels of one factor is not the same at all levels of the other factors. Box and Draper [12] demonstrate a simple means of calculating these

Table 2.1 Standard order table for 2^3 factorial design.

x_1	x_2	x_3
-1	-1	-1
+1	-1	-1
-1	+1	-1
+1	+1	-1
-1	-1	+1
+1	-1	+1
-1	+1	+1
+1	+1	+1

effects using a table of signs. The table is easily constructed by beginning with a column of 2^k +1's. The next k columns are from the standard order table labeling them **1, 2, ... , k**. Next obtain the $(2^k - k - 1)$ interaction columns **12, 13, ... , 123, ... , k** by multiplying the signs row by row as indicated by column headings. At the bottom are written the divisors, 2^k for the first column and 2^{k-1} for all the others. The effects (both main and interaction) are then calculated by adding the responses using the signs from the appropriate column and dividing by the appropriate divisor. Such a sign table for a 2^3 design is illustrated in Table 2.2. Although there are quicker means of calculating the effects [12] and in most cases analysts resort to computer programs for calculation purposes, the table of signs that they describe is useful in understanding the nature of the various factorial effects.

Table 2.2 Columns of signs and divisors for 2^3 factorial design.

Run	**I**	**1**	**2**	**3**	**12**	**13**	**23**	**123**	**y**
1	+1	-1	-1	-1	+1	+1	+1	-1	y_1
2	+1	+1	-1	-1	-1	-1	+1	+1	y_2
3	+1	-1	+1	-1	-1	+1	-1	+1	y_3
4	+1	+1	+1	-1	+1	-1	-1	-1	y_4
5	+1	-1	-1	+1	+1	-1	-1	+1	y_5
6	+1	+1	-1	+1	-1	+1	-1	-1	y_6
7	+1	-1	+1	+1	-1	-1	+1	-1	y_7
8	+1	+1	+1	+1	+1	+1	+1	+1	y_8
Divisor	8	4	4	4	4	4	4	4	

It should be pointed out that the estimated least squares coefficients (parameters) in a fitted first order polynomial graduating function are exactly one half of the main effect as defined above. The one half factor relationship occurs because the main effect measures the response change to a change of 2 units of the coded variable (-1 to +1). The regression coefficient measures the change in response when the coded variable changes by 1.

If it is possible to randomly allocate all of the experimental material, resources, and the order of the individual runs, the experimental design is considered to be fully *randomized.* It is not always possible or practicable to do this. For example, the raw material may be suspected of inhomogeneity or the runs may be accomplished on different machines or by different technicians. These differences could cause bias between the different set of circumstances, or *blocks.* By assuming that that the differences between blocks will cause the response to simply raise or lower by a fixed, unknown amount, a means of avoiding the bias is possible [12]. This is commonly referred to as blocking. Blocking introduces a new factor into the experiment, the block factor, which indicates the block that a particular run is performed. If the blocking factor is not accounted for in the experimental design, its effect will be *confounded* or *aliased* with one or more of the other effects. In other words, the analyst won't be able to tell whether the aliased effects are caused by the blocking factor or the factors aliased with the blocking factor.

The idea in blocking is to divide a factorial design into blocks of equal size and then aliasing the blocking factor with one or more of the effects that are not "important" or expected to be significant. The factor that is aliased with the blocking factor is called the *blocking generator.* Simply, the blocks are constructed by gathering the runs of the same sign of the blocking generator into the separate blocks. This can be illustrated by blocking the 2^3 factorial design in Table 2.2 into two equal blocks using the **123** factor as the blocking generator. This is shown in Table 2.3. Assuming that the blocking effect is additive, it can be shown that using this scheme, that the estimates of all of the effects, except **123**, is unchanged by the blocking effect [12]. However, the blocking effect and the **123** interaction effect are indistinguishable, or confounded. As long as the experimenter is concerned with only the main effects and the two-way interaction effects, the illustrated blocking scheme is effective. Box and Draper [12], and many other sources, provide tables for choosing block generators for 2^k factorial designs. The blocking designs are constructed so that all the main effects are not aliased with blocking effect(s) and the aliasing that does occur is with the least number of high order interaction terms.

Table 2.3 2^3 factorial design in two blocks.

BlockI

Run	**1**	**2**	**3**	**12**	**13**	**23**	**123**
1	-1	-1	-1	+1	+1	+1	-1
4	+1	+1	-1	+1	-1	-1	-1
6	+1	-1	+1	-1	+1	-1	-1
7	-1	+1	+1	-1	-1	+1	-1

BlockII

Run	**1**	**2**	**3**	**12**	**13**	**23**	**123**
2	+1	-1	-1	-1	-1	+1	+1
3	-1	+1	-1	-1	+1	-1	+1
5	-1	-1	+1	+1	-1	-1	+1
8	+1	+1	+1	+1	+1	+1	+1

Often, the analyst is concerned with reducing the number of runs. Each run costs time and money and the number of runs required for a full factorial design increases exponentially as the number of factors are increased. *Fractional* factorial designs are a method of reducing the number of experimental runs. Suppose an analyst wanted to reduce the number of runs for a 2^3 factorial design. To accomplish this, suppose the analyst performed only those runs described Block II in Table 2.3. This is called a half fraction of the 2^3 factorial, designated as a 2^{3-1} design. Notice that columns **1** and **23** are identical, along with columns **2** and **13** and columns **3** and **12**. Therefore, each of these paired effects are indistinguishable from each other. The same is true for Block I, the difference being that the signs are reversed for each confounded column. Thus, for a half fraction of the 2^3 factorial, the two-way interactions are confounded with the main effects. If it is known, or expected, that the two-way interactions are insignificant, this is a satisfactory design for estimating the main effects. If not, the full factorial design must be performed.

The *resolution* of a factorial design describes the amount of confounding within the design. To define resolution, begin with the *defining relation* which is the set of all effects (or *words*) that are equivalent to the *identity* within a fractional design. Here, identity refers to the generating effect that is composed of all "1"'s (all positive or all negative). The resolution is then defined as the length of the shortest word in the defining relation. Hence, the 2^{3-1} half fraction design described above is of resolution III, since the defining relation consist of only the **123** effect. It can be shown that that designs with the following resolution have the following properties [12]:

- Resolution III No main effects are aliased with any other main effects, but the main effects are aliased with two-factor interactions and two-factor interactions may be aliased with each other.
- Resolution IV No main effects are aliased with any other main effects or with two-factor interactions. The two-factor interactions are aliased with each other.
- Resolution V No main effects are aliased with any other main effects or with any two- or three-factor interaction effects. Two-way interaction effects are aliased with three-way interactions or higher.

Based on this, Box and Draper [12] point out that since first-order polynomials are used to construct the exploratory empirical models, designs of resolution III or higher should be used. That way, no main effect is aliased with any other main effect, although they may be aliased with two-factor interactions. If it is assumed that the model can be adequately represented by a first-order polynomial, the two-factor interactions are assumed to be zero anyway.

Many other experimental designs are available to the analyst for conducting exploratory experiments. Among the commonly used designs are foldover designs, saturated designs, and Plackett and Burman designs. Each of these are designs that are other than full fractions that meet certain aliasing requirements. Several resources are available, including Box and Draper [12], so that the analyst can design an experiment that will most efficiently fulfill the model building requirements.

2.4.4 Steepest Ascent. Up to this point, the discussion has focused on techniques for constructing first order polynomials over some portion of the operability region, as opposed to the entire region. In most cases the entire region is not explored at once, even if the full extent of the region is known. First a complex polynomial may be necessary to describe the response surface it encompasses and an excessively large number of runs would have to be performed to fit such a polynomial. Secondly, often large regions of the operability region may be known to be uninteresting or unprofitable. Since the goal of most RSM studies is to maximize (or minimize) some response, after fitting a first order polynomial in the initial region of exploration, the next step is to perform a search outside this region to locate a point where the response is maximized (or minimized), at least locally. For the rest of this discussion, only maximization will be referred to since the procedure for finding a minimum is the same, only in the opposite direction. Once this maximum is located, a new experiment can be designed and conducted to search for the global maximum. This is done with either another first order polynomial, or a second order polynomial if it becomes significant.

In Box and Draper's opinion [12], the most effective and efficient, in terms of number of runs, means of locating the local maximum is to use the method of *steepest ascent.* Given that a first order polynomial has been fitted to the data, a vector at the center of the region that makes a right angle to the planar contours of the response and points in the direction of increasing response is the direction of steepest ascent. From calculus, this vector is known as the *gradient.* The gradient points in direction of steepest ascent. For a given differentiable multivariable function f, the gradient at a point $\mathbf{a} = (a_1, a_2, \ldots, a_k)$ is defined as the vector whose elements are the partial derivatives evaluated at **a** represented by

$$\nabla f(\mathbf{a}) = \left(\frac{\partial f}{\partial x_1}(a_1), \frac{\partial f}{\partial x_2}(a_2), \ldots, \frac{\partial f}{\partial x_k}(a_k)\right)' \tag{2.68}$$

For a first order model, these partial derivative values are simply the coefficients of the main effects. A unit gradient vector is the vector of length one in the direction of the gradient vector. Although the predicted response of a fitted model is invariant to scale changes over a fixed design region, according to Box and Draper [12], the direction of the gradient does vary according to the scaling scheme selected. They point out that the steepest direction of ascent is calculated when using the units of design scaling and recommend its use.

The scheme now is to perform experiments from the center of the initial experimental design in the direction of the gradient until the maximum response on that path is observed. Some decisions that must be made include selection of appropriate step size and/or search strategy along the steepest ascent path. Another decision is whether to move in the direction of all or just the significant parameters. Finally, it must be decided whether to use the observed maximum along the path or estimate it based on a fit to the data collected. No hard and fast rules exist for these decisions. It is left to the analyst to make these decisions based on his/her best judgement based on observations of the data, past experience, and any previous knowledge of the system. As Box and Draper [12] point out, it should be remembered that

> a subject as concrete and mathematically satisfying as experimental design is actually embedded in a morass of uncertainty, uncertainty due to the possibilities that the experimenter might choose wrong variables, might explore the wrong region, or might use scaling that was inappropriate.

They do point out that all is not lost however. First, analysts often know (or can find out) a great deal about the system under study. Also, since an RSM study is conducted sequentially, the analyst doesn't have to be exactly right. Being "sufficiently" close to right will get one on one of the many possible paths to the right answer.

Other issues discussed by Box and Draper [12] include steepest ascent subject to a constraint and the confidence region for the direction of steepest ascent. Simply put, when performing a steepest ascent search and a constraint is encountered, the factor that is constrained is held at that level while the search is continued along the gradient directions for the other factors. Box and Draper [12] show that the confidence region for the gradient is represented geometrically by a cone whose vertex is at the center of the design region. A good indication that the direction of steepest ascent has been determined accurately enough is the magnitude of the solid angle of the confidence cone about the estimated vector. Box and Draper [12] provide a method for using the *t*-table for assess the size and confidence of this angle.

2.4.5 Second-Order Model Fitting. At some point in an RSM study the first-order empirical model may not adequately represent the system under study. Reasons include insignificance of the fitted first-order model, significant lack of fit, and/or significant higher order terms. Given that the model is inadequate, Box and Draper [12] describe two methods of finding an adequate model for the system. The first method involves transforming the response or predictor variables so that the system can be adequately described by a first-order model of the transformed variables. The second approach is to perform additional experimental runs in order to test for quadratic terms and then fit a second-order model if appropriate. This section briefly discusses each of these approaches, with an emphasis on the second. Finally, the section includes a discussion on experimental designs, based on two-level designs, that enable the fitting of second-order models.

Box and Draper [12] point out that the metrics (unit of measurement) used to record data are chosen for the convenience of measurement. However, the simpler metrics don't necessarily result in simpler models. Rather, some transformation of the response, predictor variables, or both may result in a simpler model. Nonlinear transformations, such as the square root, log, or reciprocal of a response effectively expand the scale in one part of the range while contracting it in another. These transformations are called the power transformations [12]. Given an appropriate transformation, a first-order model may still be used to adequately model the system. Box and Draper describe analytic (predictive score function) and graphical (residual plots) tests for determining the need for transformation. They also provide procedures for finding the optimal transformation functions.

Given that a transformation of the data doesn't result in an adequate first-order model, a simple statistical test for the presence of quadratic terms is described. By augmenting a two-level factorial design with replications at the center of the design, the sum of squares can be furthered partitioned into a sum of squares for the presence of *pure quadratic terms* [12]. The test compares the average response at the corner points of a two-level factorial design with that at the center of the design. To describe this test, let n_f represent the number of observations made at the corner points of the original 2^k or 2^{k-p} design and $n_0 > 1$ the number of observations (replications) at the center of the original design. Then the average response at the corner points can be calculated by

$$\bar{y}_{n_f} = \frac{1}{n_f} \sum_{i-1}^{n_f} y_i^{(f)} \tag{2.69}$$

where $y_i^{(f)}$ is a response from one of the corner points. The average response at the center point is found by

$$\bar{y}_{n_0} = \frac{1}{n_0} \sum_{i-1}^{n_0} y_i^{(0)} \tag{2.70}$$

with $y_i^{(0)}$ a response from the center point. If a second-order model, given by

$$E[y] = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k + \beta_{11} x_1^2 + \beta_{12} x_1 x_2 + \cdots + \beta_{kk} x_k^2 \tag{2.71}$$

is appropriate, the expected value of the average response at the corner points is given by

$$E\left[\bar{y}_{n_f}\right] = \beta_0 + \beta_{11} + \cdots + \beta_{kk} \tag{2.72}$$

and the expected value of the average response at the center point is

$$E\left[\bar{y}_{n_0}\right] = \beta_0 \tag{2.73}$$

Therefore $E\left[\bar{y}_{n_f} - \bar{y}_{n_0}\right] = \beta_{11} + \cdots + \beta_{kk}$ which are the quadratic terms. So there is an indication that the "pure" quadratic terms are important if this difference is significant.

To test for the significance the ratio SSPQ/MSPE is formed where SSPQ is the sum of squares for pure quadratic terms and MSPE is the mean square error for pure error. SSPQ is computed by

$$\text{SSPQ} = \frac{n_0 n_f \left(\bar{y}_{n_f} - \bar{y}_{n_0}\right)^2}{n_0 + n_f} \tag{2.74}$$

and has one degree of freedom. To find MSPE, the square error for pure error, SSPE is calculated by

$$\text{SSPE} = \sum_{i=1}^{m} \sum_{u=1}^{r_i} \left(y_{iu} - \bar{y}_i\right)^2 \tag{2.75}$$

which has $n - m$ degrees of freedom, where $m = n_f + 1$ is the number of design points, $n = n_f + n_0$ is the number of observations and r_i is the number of replications at the i^{th} design point. Thus MSPE $=$ SSPE$/(n - m)$.

Then if it is assumed that $E\left[\bar{y}_{n_f} - \bar{y}_{n_0}\right]$ is significant, it is known that SSPQ/MSPE has a F distribution with 1 and $n - m$ degrees of freedom. The statistical test is then formulated as

$$\begin{aligned} H_0 &: \quad \beta_{11} = \beta_{22} = \cdots = 0 \\ H_1 &: \quad \text{at least one } \beta_{ii} \neq 0 \; i = 1, 2, \ldots k \end{aligned} \tag{2.76}$$

so that if SSPQ/MSPE is greater than a value of an F distribution with degrees of freedom 1 and $n - m$, the null hypothesis is rejected.

Next, the focus is to build on, or *augment* the two-level factorial designs for first-order models in order to fit a full second-order model. Although the center point replications from the procedure described above, permit investigation into model curvature, the additional replications don't permit for estimation of a full quadratic model [12]. A central composite design (CCD) builds on the two-level factorial design that has been augmented with center point replications. A CCD is constructed by adding $n_a = 2k$ *axial* design points of the form

$$(\pm\alpha, 0, \ldots, 0), \;\; (0, \pm\alpha, \ldots, 0) \; (0, 0, \ldots, \pm\alpha) \tag{2.77}$$

where α is usually equal to $n_f^{1/4}$, with n_f the number of design points in the factorial portion of the design. The CCD has several properties to recommend it, in addition to having an embedded two-level factorial design. The design is a rotatable second-order minimum bias design. Rotatable means that the accuracy of the predicted response is a function of only the distance from the center of the design, not the direction [12].

Unless the analyst expects that a second-order model will be required, runs at the axial design points may be performed after the results from a two-level factorial design are analyzed. In that case, a blocking effect is likely the result of the sequential experimentation. In that case, it is possible to design a CCD where the block effects can be estimated separately and independently from those of the other factors. This type of design is said to *blocked orthogonally* and is accomplished by the selection of α and the number of center point replications performed in each block [12]. Since this dissertation is concerned with performing RSM on a simulation model, it is not necessary to be concerned with this type of blocking as long as all replications are independent.

2.4.6 Exploration of Maxima and Ridge Systems. Given a fitted second-order model, the next task is to search for a maximum. Box and Draper [12] describe a general strategy using canonical analysis that is summarized here. Canonical analysis is an approach for analyzing the fitted second-order model by rotating the axes to remove all cross-product terms and when necessary translation of the coordinate axes to coincide with the stationary point. A model that has been rotated and translated in this manner is said to be in canonical form. The advantages of this form allow the analyst to identify a local optimum, if one exists, and to describe the response surface in a straightforward manner. When the model is only rotated, that is referred to *A canonical form.* A rotated and translated model is said to be in *B canonical form.*

The following matrix notation is developed for use throughout this section. A quadratic function given by

$$y = b_0 + b_1x_1 + \cdots + b_kx_k + b_{12}x_1x_2 + \cdots + b_{kk}x_k^2 \tag{2.78}$$

can be represented in matrix form as

$$y = b_0 + \mathbf{x}'\,\mathbf{b} + \mathbf{x}'\,\mathbf{B}\mathbf{x} \tag{2.79}$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_k)'$, $\mathbf{b} = (b_1, b_2, \ldots, b_k)'$, and

$$\mathbf{B} = \begin{bmatrix} b_{11} & \frac{1}{2}b_{12} & \cdots & \frac{1}{2}b_{1k} \\ \frac{1}{2}b_{12} & b_{22} & \cdots & \frac{1}{2}b_{2k} \\ \vdots & \vdots & \ldots & \vdots \\ \frac{1}{2}b_{1k} & \frac{1}{2}b_{2k} & \ldots & b_{kk} \end{bmatrix} \tag{2.80}$$

From calculus, it is known that a necessary condition for a point to be a maximum (minimum), it must be a *stationary point.* For a point to be a stationary point, the first derivative (if it exists) evaluated at that point equals zero. For the second-order model described here, a stationary point can be found (if it exists) by setting the derivative of y equal to the $k \times 1$ zero vector and solving. Since the derivative of y is given by

$$\frac{dy}{d\,\mathbf{x}} = \mathbf{b} + 2\,\mathbf{x}'\,\mathbf{B} \tag{2.81}$$

the stationary point, if it exists, is found by solving

$$\mathbf{x}_s = -\frac{1}{2}\,\mathbf{B}^{-1}\,\mathbf{b} \tag{2.82}$$

.

If a stationary point exists, the next step of the general strategy is to determine its Euclidean distance from the center point of the current experimental design. This is computed as

$$D = \sqrt{\sum_{i=1}^{k}(x_S)_i} \tag{2.83}$$

Based on D, determine if $\mathbf{x}_s$ is within the current experimental design space. If it is, Box and Draper [12] recommend that the empirical model be transformed to B canonical form in order to characterize the response surface. B canonical form is given by

$$y = y_S + \tilde{\mathbf{X}}' \, \Lambda \, \tilde{\mathbf{X}} \tag{2.84}$$

where Λ is the $k \times k$ diagonal matrix whose non-zero elements are the eigenvalues of $\mathbf{B}$ and y_S is the value of the response at the stationary point (origin of the new coordinate system). $\tilde{\mathbf{X}} = \mathbf{M}'(\mathbf{x} - \mathbf{x_s})$ where x_s is the stationary point and $\mathbf{M}$ is $k \times k$ orthonormal matrix whose columns are the standardized eigenvectors of $\mathbf{B}$.

If D indicates that the stationary point is outside of the experimental design space, or if no stationary point exists, Box and Draper recommend putting the model in A canonical form in order to characterize the response surface. This is accomplished via

$$y = b_0 + \boldsymbol{\theta}' \, \mathbf{X} + \mathbf{X}' \, \Lambda \mathbf{X} \tag{2.85}$$

where $\mathbf{X} = \mathbf{M}' \, \mathbf{x}$ and $\boldsymbol{\theta} = \mathbf{M}' \, \mathbf{b}$. $\mathbf{M}$ and Λ are the same as defined above. With the model in A canonical form, the next step is to perform a search for the optimum by performing either *elucidation of a ridge system* or *ridge analysis.*

A stationary point in a second-order model far from the center of an experimental design region implies the existence of a ridge system [12]. Elucidation of a ridge system is an analytical means of detecting, describing and exploiting the ridge system. To perform elucidation of a ridge system, after transforming the model to A canonical form, the analyst studies four measures for each factor. These measures are

- θ_i slope for factor i
- λ_i quadratic coefficient for factor i
- X_{iS} distance from the design center to the stationary point for factor i

- r_i approximate range of $\hat{y}$ due to factor i over the design region with $r_i = \sqrt{3\theta_i^2 + \lambda_i^2}$

It should be pointed out that each of these measures are statistical estimates of the true values. Since the CCD design is orthogonal and rotatable, the standard error of the original, coded regression coefficients can be used for an approximation of the standard errors of the A canonical coefficients [12]. Based on these measures, the ridge is estimated and a direction along the ridge is determined for further experiments. The details of the method are outlined in Box and Draper [12].

Ridge analysis locates the maximum estimated response that is a distance R from the design center. The method can be posed as the following constrained optimization problem

$$\begin{array}{ll} \text{Maximize} & \hat{y} = b_0 + \mathbf{x}'\,\mathbf{b} + \mathbf{x}'\,\mathbf{B}\mathbf{x} \\ \text{subject to} & \sum_{i=1}^{k} x_i^2 - R^2 = 0 \end{array} \qquad (2.86)$$

Although Box and Draper [12] don't recommend its use, several computer packages, including SAS, make the method available. In the computer packages, ridge analysis for many values of R can be rapidly performed. The result is that a new design center can be located for further experimentation.

III. Simulation and Analytic Modeling

3.1 Overview

Throughout this dissertation, we use both simulation and analytical models in concert to analyze and study several different systems. Because the methods described often commingle the inputs and outputs of both types of models we have included a short discussion on the two types of models. For clarity, the notation developed here will be used throughout the rest of the dissertation. We begin by defining the terms model and system for the purposes of this dissertation. This is followed by a discussion of discrete event simulation models with a short tutorial on one method of estimating output performance measures using simulation models. We then present a discussion on analytical models.

3.2 Systems and Models

Essentially, a system is the "thing" under study. According to Law and Kelton [32], "A system is defined to be a collection of entities, e.g., people, or machines, that act and interact together toward the accomplishment of some logical end. ... In practice, what is meant by 'the system' depends on the objectives of a particular study." Pritsker [40] describes a system as "a collection of items from a circumscribed sector of reality that is the object of study or interest." Balci states simply, the term "system is used to refer to the entity that contains the problem to be solved" which can have inputs, parameters, and outputs [5].

We adopt the simple definition offered by Balci [5], which is depicted graphically in Figure 3.1. The roughness of the "system" block implies that the boundary between the system and the rest of reality is not easily defined and depends on the problem statement and the judgement of the analyst. The inputs to the system include a vector of structural parameters $\boldsymbol{\phi} = (\phi_1, \phi_2, \cdots, \phi_i)'$ that consists of those things input to the system that don't change over time or space. For example, elements within $\boldsymbol{\phi}$ could include the number and types of aircraft in a particular fleet of aircraft or the number of refueling hydrants at a particular air base. The variable input vector, $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_j)'$, contains those inputs to the system that do change over time or space. Some examples of input variables include the time required to service an aircraft or the amount of fuel stored at a particular air base. The output vector $\mathbf{Y} = (Y_1, Y_2, \dots, Y_k)'$ includes values that may or may not change over time or space and are the result of the system acting upon the different inputs. Each element of the output vector could be a vector of several occurrences of the item under interest or could be single value. Examples of outputs include aircraft throughput or the times required to service aircraft over a specified amount of time.

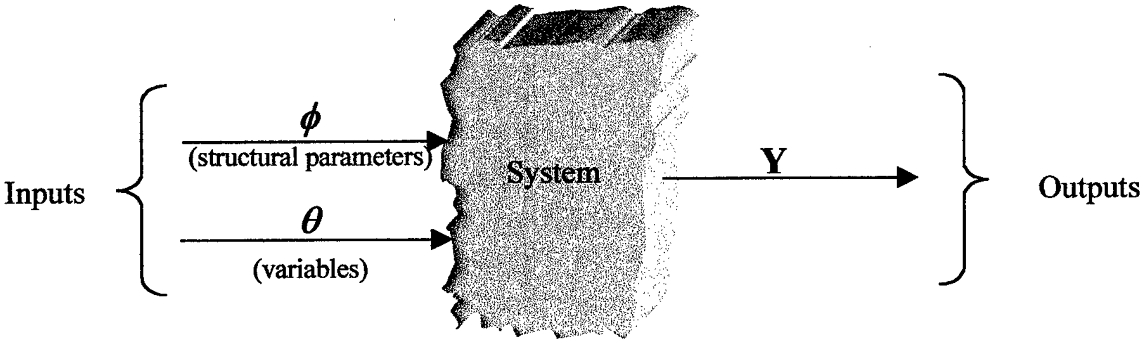

Figure 3.1 System characteristics.

As with the term system, there are several different definitions for a model. First we note that we are referring to mathematical models, not physical models. A mathematical model represents "a system in terms of logical and quantitative relationships that are then manipulated and changed to see how the model reacts, and thus how the system *would* react—*if* the model is a valid one [32]." More simply, a model "is a representation of a system with inputs, parameters, and outputs [5]." There are several different types of mathematical models. In the next sections we focus on two of them—simulation and analytical models.

3.3 Simulation Models

For this dissertation, simulation model refers to a discrete event simulation model. Such models are *dynamic* in that they represent a system as it evolves over time, *discrete* in that the state variables that describe the system change instantaneously at separated points in time, and *stochastic* in that they have random inputs and outputs [32]. Due to the size and complexity of most simulation models, they are normally translated to computer code so that they can be evaluated on a computer. One approach to defining a simulation model is depicted in Figure 3.2. In this case we choose to define a simulation model in an analogous manner to the definition of a system provided in Figure 3.1.

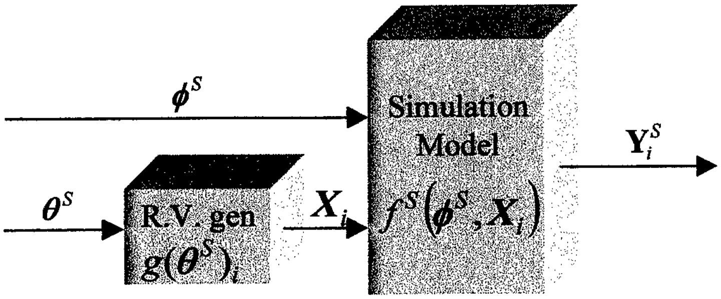

Figure 3.2 Simulation model characteristics

We define a simulation model as the multivariable function, $f^S\left(\phi^S, g\left(\theta^S\right)_i\right) = \mathbf{Y}_i^S$, where the superscript S is used to identify the function, inputs, and outputs as part of the simulation model. As before, ϕ^S is a vector of structural parameter inputs that in this case do not change during a realization of the simulation model. For example, these structural parameters might represent the number of aircraft or parking spots in the system being modeled. The random variable parameter vector θ^S consists of the parameters that specify the random variables that drive the simulation model. These random variates could represent service times or failure rates for example. For clarity, we have located the random variate generator (actually psuedo-random variate generator) outside of the simulation model. The random variate generator is a separate function, g, of the random variate parameter vector, θ^S that produces an output vector, $\mathbf{X}_i$, for replication i whose elements are independently and identically distributed (IID) stochastic processes. Hence, the random variate generator is given by $g\left(\boldsymbol{\theta}^S\right)_i = \mathbf{X}_i$. The simulation model then transforms these inputs into a vector of output stochastic process for replication i represented by $\mathbf{Y}_i^S$. These stochastic processes could include the set of all observed repair times or the times that aircraft depart an airfield. Note that most of the output stochastic processes are not independently nor identically distributed [32]. Included in our definition of $\mathbf{Y}_i^S$ are the realized random variate stochastic processes that are also inputs to the model (which are IID).

Since the outputs of a simulation model are stochastic processes that are not IID, several methods of simulation output analysis have been developed. We present the method of independent replications used throughout this dissertation. The following presentation is excerpted from Law and Kelton [32] and is graphically depicted in Figure 3.3. To implement this method, n independent replications of the simulation model are generated. By independent replication, we mean that $\mathbf{X}_i$ is statistically independent of $\mathbf{X}_k$ for $i = 1, 2, \ldots, n$; $k = 1, 2, \ldots, n$; and $i \neq k$. Assume that we are attempting to estimate the mean of a single element, or performance measure of $\mathbf{Y}_i^S$ given by $E[Y_P^S] = \mu$. Then let $\mathbf{Y}_{P(i)}^S = (Y_{P(i1)}^S, Y_{P(i2)}^S, \cdots, Y_{P(im)}^S)'$ be the realized output stochastic process for the performance measure of interest for replication i. We begin by finding the sample mean of each replication by

$$\bar{Y}_{P(i)}^S = m^{-1} \sum_{j=1}^{m} Y_{P(ij)}^S \qquad i = 1, 2, \ldots, n \tag{3.1}$$

This is shown in Figure 3.3 in the "Replication" block of the "Output Analysis" section. Then we take the overall mean in order to estimate μ by

$$\hat{\mu} = \bar{Y}_P^S = n^{-1} \sum_{i=1}^{n} \bar{Y}_i^S \tag{3.2}$$

This step is depicted in the "Overall" output analysis block in Figure 3.3. Since each replication is independent of the others, each $\bar{Y}^S_{P(i)}$ are also independent of each other. Therefore, usual statistical methods can be applied to $\hat{\mu}$ to determine confidence intervals or other statistical tests. We also note that in some cases, a number of realizations at the beginning of the output stochastic process may be deleted if the analyst is attempting to estimate a steady state value [32].

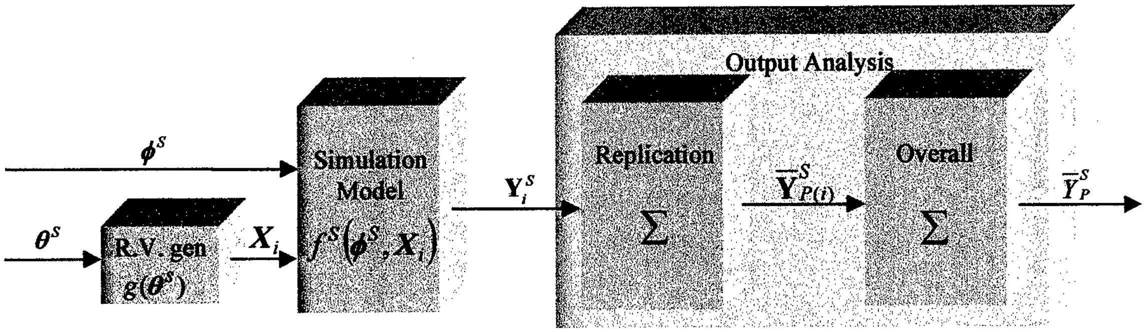

Figure 3.3 Simulation model method of independent replications.

We also graphically display the internal control variate method of variance reduction for independent replications in Figure 3.4. The control variate method is presented in some detail in Section 2.2.1 so we keep our comments brief here. Mainly we wish to point out that the vector of internal control variates, $\mathbf{C}_i$ are actually elements of $\mathbf{Y}_i^S$. As described earlier, the realized input stochastic processes, such as service times at a particular service center, can be collected as an output process for use as an internal control variate. Also note that the control variate method is a "post-processing" method in that it is accomplished after all replications are generated.

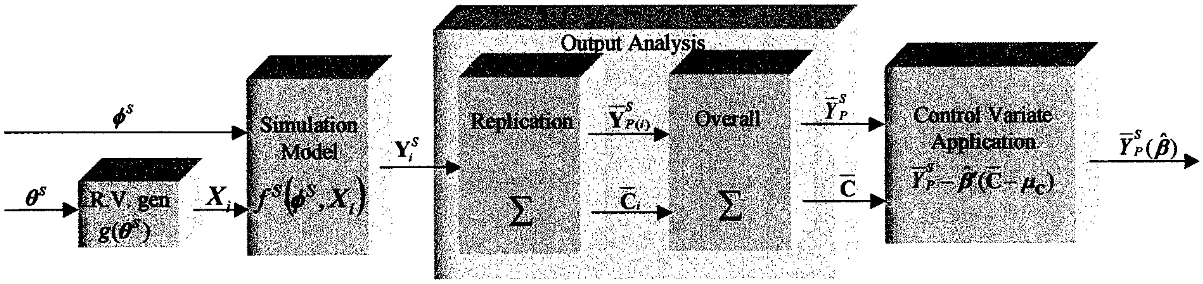

Figure 3.4 Simulation model method of internal control variates.

3.4 Analytical Models

Unlike simulation models, analytical models provide exact, *analytical* solutions [32]. A simple example of an analytical model is given by the formula $V = IR$ where V is the voltage, I the current, and R the resistance in a series electrical circuit. Other analytical models can be extremely complex and require a computer to reach a solution in a reasonable amount of time. All analytical models used in this dissertation are models of closed queueing networks. The models vary in complexity, but all require the use of a computer to calculate solutions.

We graphically depict an analytical model of the type used in this dissertation in Figure 3.5. As with the simulation model, an analytical model can be defined as the function $f^A\left(\phi^A, \theta^A\right) = \mathbf{Y}^A$ where the inputs to the model are separated into two separate vectors. The vector of structural parameters, ϕ^A, defines the structural, or topological elements that define the network modeled. Examples include the number of customers, customer classes, and the number of service center, or queues, in the network. The random variate parameter vector, θ^A define the parameters, or moments, of the random variates modeled. These might be the mean service times at each of the service centers or the probabilities associated with the routing of customers throughout the network. The output vector $\mathbf{Y}^A$ of this type of model consists of probabilities and/or mean values for any number of performance measures. Station throughput or the probability of 4 customers at a certain service center are examples of analytical model output.

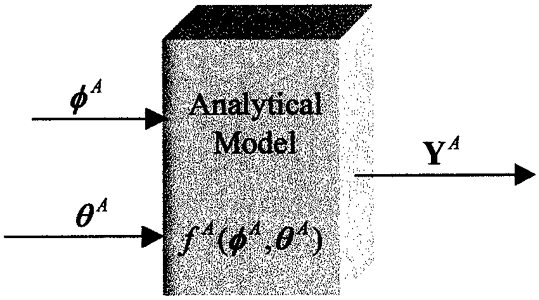

Figure 3.5 Analytical model characteristics.

IV. Analytic Control Variate Monte Carlo Method

4.1 Overview

The control variate method is a well-known variance reduction technique for discrete event simulation. This chapter explores the variance reduction achieved by employing an external analytical model to consolidate multiple input random variables into a single "analytical" control variate (ACV). As discussed in Section 2.2.5, previous researchers have found that this approach can produce significant variance reduction, but the resulting point estimate of the simulation response may exhibit unacceptable bias. In this chapter a general Monte Carlo method of distribution sampling for resolving the bias problem is developed and demonstrated through a queueing network example. The mean and variance of the input random variable used to produce the ACV must be known in order to apply the Monte Carlo method. Additionally, the distribution of these random variables must be known either exactly or approximately. In many cases, this requirement is not difficult to meet. With this modification, the ACV method performs favorably when compared with the classical internal and external control variate approaches. To demonstrate, the different control variates are compared using confidence interval width, realized coverage, and estimated mean square error (MSE) using a queueing network example. The efficiency of the ACV method is also compared to that of the uncontrolled response.

This chapter is organized in the following manner. It begins with a discussion of the ACV method and a description of the Monte Carlo method for reducing bias. Included is a discussion on the efficiency of the method. This is followed by a description of the queueing network and different control variates used to compare the different methods. Finally a comparison of the results for each of the control variates is presented.

4.2 Analytical Control Variates

4.2.1 ACV Construction. Previous researchers have introduced and explored the concept of an ACV [18, 34, 48, 49, 53, 54]. This variance reduction approach can be considered a hybrid of both internal and external methods since an external analytical model is employed to produce a new random variable that is essentially a function of internal (input) random variables. An ACV, Z, is generated for every replication of a simulation model using the following general scheme. For each replication, compute the sample means of the realized input random variables of the simulation model. Then compute the value of the analytical model using those sample means. That analytical model value is the ACV for that particular replication.

The simulation and analytical models are described as follows. Recall that we define a simulation model, for replication j, as the function

$$f^S\left(\boldsymbol{\phi}^S, \mathbf{X}_j\right) = f^S\left(\boldsymbol{\phi}^S, g\left(\boldsymbol{\theta}^S\right)_j\right) = \mathbf{Y}_j^S \qquad j = 1, 2, \ldots, n \tag{4.1}$$

where $\boldsymbol{\phi}^S$ is the vector of structural parameters, $\boldsymbol{\theta}^S$ is the vector of random variate parameters, $g(\cdot)$ is the random variate generator, and $\mathbf{X}_j$ is the vector of IID stochastic processes generated during the j^{th} replication by the random variate generator. We consider a simulation model where we wish to estimate an unknown performance measure, $E[Y_P^S] = \mu_P$ that is a function of m input stochastic processes given by $\mathbf{X}_j = (\mathbf{X}_{1j}, \mathbf{X}_{2j}, \ldots, \mathbf{X}_{mj})'$ for replication j. Then the i^{th} stochastic process generated during replication j is given by $\mathbf{X}_{ij} = (X_{ij1}, X_{ij2}, \ldots, X_{ijr_i})'$, where r_i is a constant dimension determined by the simulation replication stopping rule. Since each $\mathbf{X}_{ij}$ is an IID stochastic process, we let each X_{ijk} be a realization of the random variable X_i for $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n$, and $k = 1, 2, \ldots, r_i$. Assume that the probability structure of each X_i, including the values $E[X_i] = \mu_i$ and $Var(X_i) = \sigma_i^2, i = 1, 2, \ldots, m$ is known.

The response yielded by an appropriate analytical model of the same stochastic system is essentially a function of the moments of a subset of the same input random variables (ideally, all of the "influential" inputs). The analytical model is represented as the function $f^A(\boldsymbol{\phi}^A, \boldsymbol{\theta}^A) = Z$, where Z is the same system performance measure estimated by $E[Y_P^S]$, and $\boldsymbol{\phi}^A$ and $\boldsymbol{\theta}^A$ are the input structural and random variate parameter vectors. If the analytical model is a reasonably accurate representation of the system under study, then the analytical response should be highly correlated with the simulation response.

For example, consider simulation and analytical models of a fast food drive-through system with two windows in series. Using a discrete event simulation model, an appropriate value for $E[Y_P^S]$ might be the steady-state mean sojourn time experienced by a random customer. The random variables X_1 and X_2 might represent the customer service times experienced at the two windows. Additionally, X_3 might represent the customer interarrival time. The corresponding analytical model might employ a queueing network algorithm to calculate Z, the analytical mean sojourn time. Appropriate moments of X_1, X_2 and X_3 would serve as inputs to the analytical model.

An ACV, Z_j, is constructed for each replication j using the sample first moment estimators of some subset of the realized observations of $\mathbf{X}_{1j}, \mathbf{X}_{2j}, \ldots, \mathbf{X}_{mj}$. Rather than using the input stochastic processes themselves, we consider the simulation output processes $\mathbf{Y}_{A(j)}^S = (\mathbf{Y}_{A(1j)}^S, \mathbf{Y}_{A(2j)}^S, \ldots, \mathbf{Y}_{A(mj)}^S)'$ such that $\mathbf{Y}_{A(ij)}^S \subseteq \mathbf{X}_{ij}$ for $i = 1, 2, \ldots, m$. The subscript A denotes those outputs of the simulation model that are used as inputs to the analytical model for the purposes of generating an ACV. We make this distinction between simulation inputs and outputs for two reasons. First, when estimating steady state simulation models using the replication/deletion method, a certain number of realizations are deleted before statistics are computed so that we are not operating on the full input processes. Secondly, in the next chapter we consider inputs to the analytical model for generating an ACV that are only simulation outputs and don't have a corresponding stochastic process input. To generate ACV's perform n independent replications of the simulation. For each replication j, the simulation produces a vector of output realizations for each of the m input random variables. Represent these realized vectors by $\mathbf{Y}^S_{A(ij)} = (Y^S_{A(ij1)}, Y^S_{A(ij2)}, \ldots, Y^S_{A(ijr_i)})'$; $i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, n$ and r_i is a constant dimension determined by the simulation stopping rule. Form a new vector, $\bar{\mathbf{Y}}^S_{A(j)} = \left(\bar{Y}^S_{A(1j)}, \bar{Y}^S_{A(2j)}, \ldots, \bar{Y}^S_{A(mj)}\right)'$ for each replication j, where

$$\bar{Y}^S_{A(ij)} = r_i^{-1}\mathbf{1}'\mathbf{Y}^S_{A(ij)} \quad i = 1, 2, \ldots, m;\ j = 1, 2, \ldots, n \tag{4.2}$$

Then each $\bar{\mathbf{Y}}^S_{A(j)}$ is used as an input to the analytical model, so that $f^A(\phi^A, \bar{\mathbf{Y}}^S_{A(j)}) = Z_j$ is the ACV for replication j. If $E\left[f^A(\phi^A, \bar{\mathbf{Y}}^S_{A(j)})\right] = \mu_Z$ is known, we construct the control variate estimator as before. The analytically controlled estimate of μ is defined as

$$\bar{Y}^S_P(\hat{\beta}) = \bar{Y}^S_P - \hat{\beta}\left(\bar{Z} - \mu_Z\right) \tag{4.3}$$

where $\bar{Z}$ is the sample mean of the ACVs for all n replications and $\hat{\beta}$ is estimated as in Equation (2.15). As long as Z and Y^S_P are strongly correlated, significant variance reduction will occur.

4.2.2 Monte Carlo Method. Finding the value of μ_Z to construct an unbiased estimate of μ is often a difficult problem. If the analytical model represents a linear function, then the linearity of the expectation operator guarantees that $\mu_Z = E\left[f^A(\phi^A, \bar{\mathbf{Y}}^S_{A(j)})\right] = f^A\left(\phi^A, (\mu_1, \mu_2, \ldots, \mu_m)'\right)$. Since the relationship is normally non-linear, computation of μ_Z in this manner will produce a biased result for both μ_Z and the analytically controlled estimate of μ. An alternative numerical approximation (say $\hat{\mu}_Z$) must be found in this case. It is shown that given a known probability structure for the input random variables, $\hat{\mu}_Z$ can be accurately determined through a Monte Carlo approach.

The value of μ_Z is estimated using a general Monte Carlo method that generates a series of random vectors that approximate (in distribution) the inputs that the analytical model sees when generating the ACV. By generating enough of these vectors and calculating the value of the analytical model evaluated at each of these vectors, an accurate approximation of μ_Z can be obtained. Recall that for every replication j, the elements of each input vector $\mathbf{Y}^S_{A(ij)} = (Y^S_{A(ij1)}, Y^S_{A(ij2)}, \cdots, Y^S_{A(ijr_i)})'$ have known expectation μ_i and known variance σ_i^2. From this, the input to the analytical model for replication j, $\bar{\mathbf{Y}}^S_{A(j)}$, is formed as the vector of sample means of $\mathbf{Y}^S_{A(ij)}$. Thus, $E\left[\bar{\mathbf{Y}}^S_{A(j)}\right] = (\mu_1, \mu_2, \cdots, \mu_m)'$, for $j = 1, 2, \ldots, n$. If each X_i is reasonably large, the *central limit theorem* can be applied to each element of $\bar{\mathbf{Y}}^S_{A(j)}$. Therefore, the distribution of each $\bar{Y}^S_{A(ij)}$ can be approximated with a normal distribution having mean μ_i and variance σ_i^2/r_i (where r_i is a constant dimension of $\mathbf{Y}^S_{A(ij)}$, determined by the simulation stopping rule).

To calculate μ_Z, first generate G Monte Carlo vectors

$$\mathbf{Y}^{MC}_{A(j)} = \left(Y^{MC}_{A(1j)}, Y^{MC}_{A(2j)}, \cdots, Y^{MC}_{A(mj)}\right)' : j = 1, 2, \ldots, G \tag{4.4}$$

where each $Y^{MC}_{A(ij)}$ is an independent pseudo-random variate from a normal distribution having mean μ_i and variance σ_i^2/r_i. Then approximate μ_Z with

$$\hat{\mu}_Z = \frac{1}{G}\sum_{j=1}^{G} f^A\left(\phi^A, \mathbf{Y}^{MC}_{A(j)}\right) \tag{4.5}$$

The ACV Monte Carlo method is depicted in Figure 4.1 in the same manner as that presented in chapter III. Given that $f^A(\cdot)$ is non-linear, $\hat{\mu}_Z = \mu_Z$ as $G \to \infty$ if and only if each $\mathbf{Y}^{MC}_{A(j)}$ is sampled from the exact distribution of $\bar{\mathbf{Y}}^S_{A(j)}$. If the random variables X_i, that define the stochastic processes $\mathbf{Y}^S_{A(ij)}$, are not independent, any of several well-known techniques for generating dependent random variates may be applied [32] . One particular approach for generating dependent variables is demonstrated later in this chapter.

4.2.3 Monte Carlo Method Efficiency. Determining the optimal value of G involves a trade-off between accuracy and computational efficiency. To achieve a desired level of accuracy, one can apply standard statistical techniques to determine the number of replications necessary to

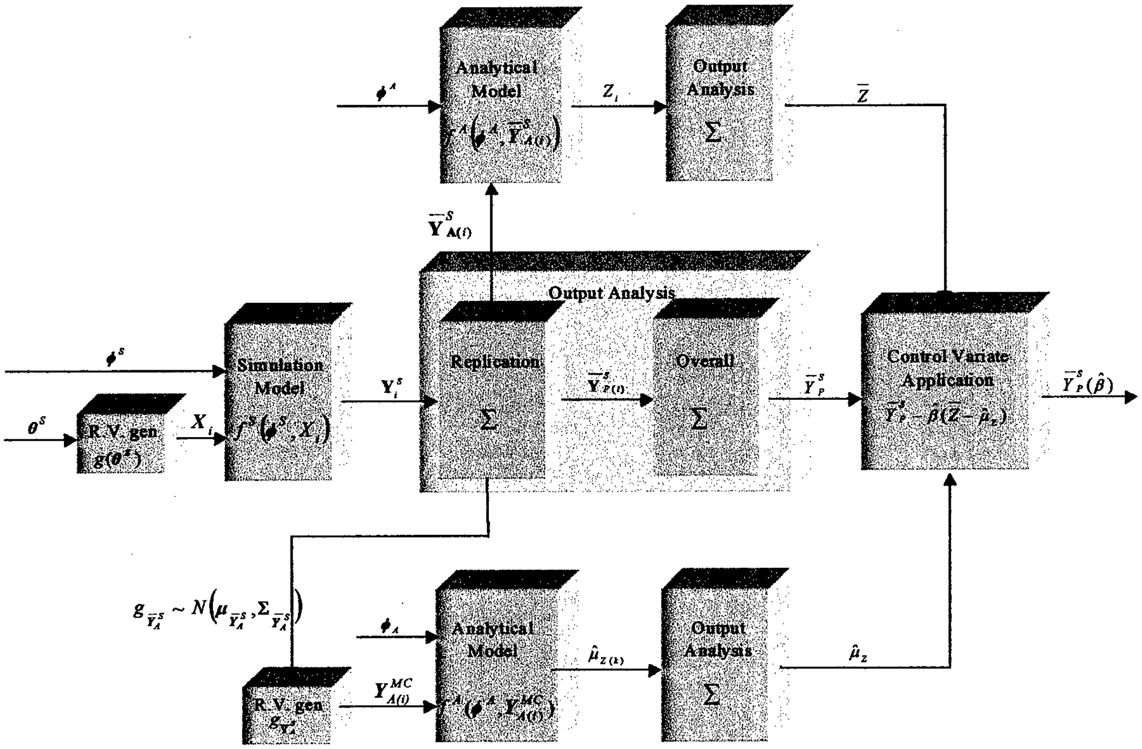

Figure 4.1 ACV Monte Carlo method of variance reduction.

construct a confidence interval of any width about an estimate of μ_Z. The smaller this confidence interval, the less likely the resulting value of $\hat{\mu}_Z$ will cause excessive bias in the ACV-controlled response estimate. Based on the results from this chapter, no detectable bias will occur if the confidence interval width (CIW) about $\hat{\mu}_Z$ is less than 10 percent of the CIW about the ACV-controlled response (for typical values of α). However, as more computer time is expended calculating $\hat{\mu}_Z$, less time is available for simulation replication and ACV production. Consider the following questions. Is a better estimate of μ obtained by simply allocating all available computer time to replicating the simulation and forgoing the use of the ACV? If not, what value of G provides a sufficiently accurate estimate of μ_Z while allowing enough time to produce a sufficiently narrow CIW for the ACV-controlled response? The answers to these questions depend on several factors that are unknown before the simulation is replicated. These factors include the times required to produce a simulation replication, a single ACV, and a single Monte Carlo estimate of μ_Z. Other factors to be considered include the amount of variance reduction achieved using the ACV, and the comparative widths of the confidence intervals about the ACV-controlled response and $\hat{\mu}_Z$. Though the issues may seem complex, sufficient insight can be obtained in a straightforward manner.

To determine a satisfactory value for G, begin by allocating some fraction of the available computer time to perform a small pilot study and obtain estimates of all the applicable factors. Consider the case where a fixed amount of computer time, $\mathcal{T}$, is allocated for a complete simulation study. The goal is to determine a satisfactory value for G that is both accurate and efficient. To accomplish this, compare the widths of the predicted confidence intervals for the uncontrolled and ACV-controlled responses, based on the pilot study values. Label the times required to produce a simulation replication, and ACV, and a Monte Carlo replication as t_{REP}, t_{ACV}, and t_{MC} respectively. The maximum possible number of uncontrolled replications is given by

$$n_{UNC} = \left\lfloor \frac{\mathcal{T}}{t_{REP}} \right\rfloor \tag{4.6}$$

where $\lfloor i \rfloor$ returns the nearest integer less than or equal to i. The predicted $(1-\alpha)100\%$ CIW for the uncontrolled response is given by

$$CIW_{UNC} = 2t_{1-\alpha/2, n_{UNC}-1}\sqrt{\frac{S_Y^2}{n_{UNC}}} \tag{4.7}$$

where S_Y^2 is the sample variance of Y. To predict the $(1-\alpha)100\%$ CIW for the ACV-controlled response, CIW_{ACV}, we select an initial value of G. The maximum possible number of simulation replications using an ACV is then given by

$$n_{ACV} = \left\lfloor \frac{\mathcal{T}}{t_{REP} + t_{ACV} + Gt_{MC}} \right\rfloor \tag{4.8}$$

An initial estimate of CIW_{ACV} is obtained using Equation (2.8) to estimate $Var\left[\bar{Y}\left(\hat{\beta}\right)\right]$, by

$$CIW_{ACV} = 2t_{1-\alpha/2,n_{ACV}-2}\sqrt{\frac{n_{ACV}-2}{n_{ACV}-3}\left(1-r_{YZ}^2\right)\frac{S_Y^2}{n_{ACV}}} \tag{4.9}$$

where r_{YZ}^2 is the maximum likelihood estimator of the square of the correlation coefficient between Y and Z (the ACV). Then adjust iteratively, re-computing Equations (4.8) and (4.9) until the CIW about $\hat{\mu}_Z$ is approximately 10% of CIW_{ACV}. Then if CIW_{ACV} is less than CIW_{UNC}, the ACV-controlled response estimate is more accurate than the uncontrolled estimate. Otherwise it is more efficient to forego the application of the ACV. Note two items. First, it is not claimed that Equation (4.9) is an exact formula, but only that it provides a useful estimate. Secondly, it is possible to arrive at a value of G that is both efficient and provides a CIW about $\hat{\mu}_Z$ that is much narrower than 10% of CIW_{ACV}. It is left to the best judgement of the simulation practitioner to determine a satisfactory value G in that case.

4.3 Queueing Network Example

The performance of the various control variate methods can be compared using the example queueing network example shown in Figure 4.2. Lavenberg, Moeller, and Welch [30] have previously examined this classic model within a variance reduction context. The queueing network is composed of S service stations with N customers circulating between them. Call this model Q_1. Station 1 has exactly N servers, resulting in no queueing at station 1. The remainder of the stations are single server queues, all employing a first come-first served service discipline. The transition probability matrix for customer movement between stations is given by

$$\mathbf{P} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ p_1 & 0 & p_3 & \cdots & p_S \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 1 & 0 & \cdots & 0 \end{bmatrix} \tag{4.10}$$

with $\sum_{i=1}^{S} p_i = 1$. Any customer completing service at station $k = 1, 3, \dots, S$ is immediately routed to station 2. Upon completion of service at station 2, the customer is routed to station k with probability p_k. Station 2 has an exponential service time distribution with mean λ. Each of the other stations has a Weibull service time distribution with shape parameter $\alpha_k > 0$ and scale parameter $\beta_k > 0$. The mean service time for each station is therefore

$$\mu_k = \beta_k^{1/\alpha_k} \Gamma\left(\frac{\alpha_k + 1}{\alpha_k}\right), \quad k = 1, 3, \dots, S \tag{4.11}$$

and the variance is

$$\sigma_k^2 = \beta_k^{2/\alpha_k} \left\{ \Gamma\left(\frac{\alpha_k + 2}{\alpha_k}\right) - \Gamma\left(\frac{\alpha_k + 1}{\alpha_k}\right)^2 \right\}, \quad k = 1, 3, \dots, S \tag{4.12}$$

where $\Gamma(\cdot)$ is the gamma function.

Model Q_1 is a simple representation of an interactive multiprogrammed computer system. The customers in the network are users of the system, with station 1 representing the user terminals. Each service time at station 1 represents a user's "think" time between system task requests. Station 2 represents the system's central processing unit (CPU) and stations $3, \dots, S$ denote mass storage units (disk, drum, tape, etc.). The service time at station 2 represents processing time until either a task is completed (in which case the customer returns to station 1) or data from a mass

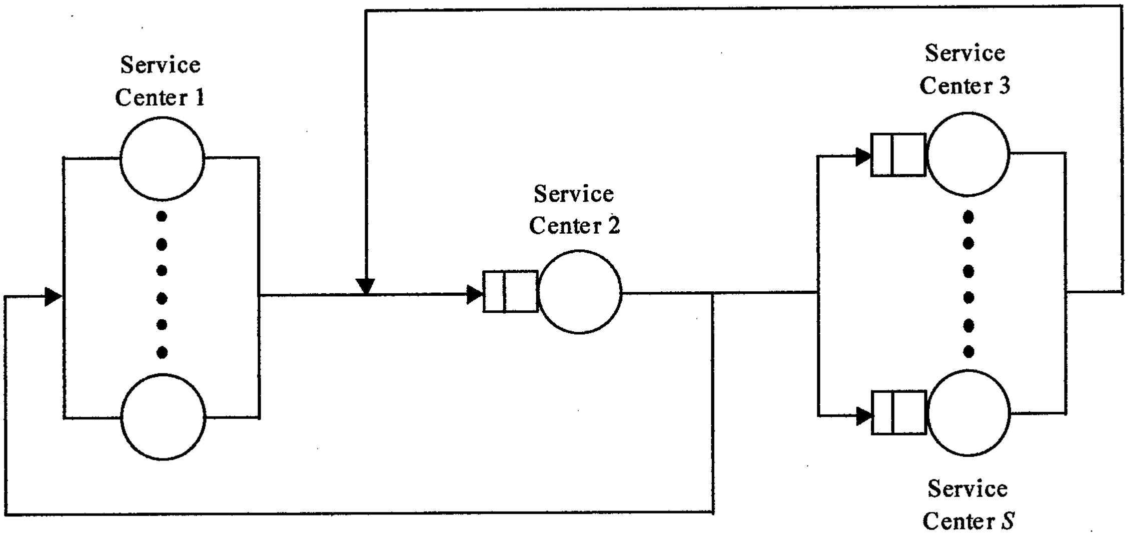

Figure 4.2 Closed queueing network Q_1

storage unit is required. A service time at stations $3, \ldots, S$ represents the time required to transfer data from the storage device to the main memory, where it can be acted upon by the CPU.

Users must be allocated a portion of main memory in order to access the CPU and storage devices. Since memory is limited, all users may not be permitted memory access at the same time. This leads to a variation of Q_1 (called Q_2), that is shown in Figure 4.3. In model Q_2, at most $N' < N$ customers can enter the subnetwork comprised of stations $2, \ldots, S$. A new queueing station H holds customers in delay until the customer population in the subnetwork is less than N'. The service time for station H is zero and no queueing occurs at H if the customer population in the subnetwork is less than N'.

Numerous performance measures may be of interest for these notional computer systems, but two measures are particularly important. The first is the system sojourn time, defined here as the long-run average time between a customer's visits to station 1. The other measure is the steady-state CPU utilization (the long-run fraction of time station 2 is busy serving customers). These measures respectively address effectiveness and efficiency and are of interest for both Q_1

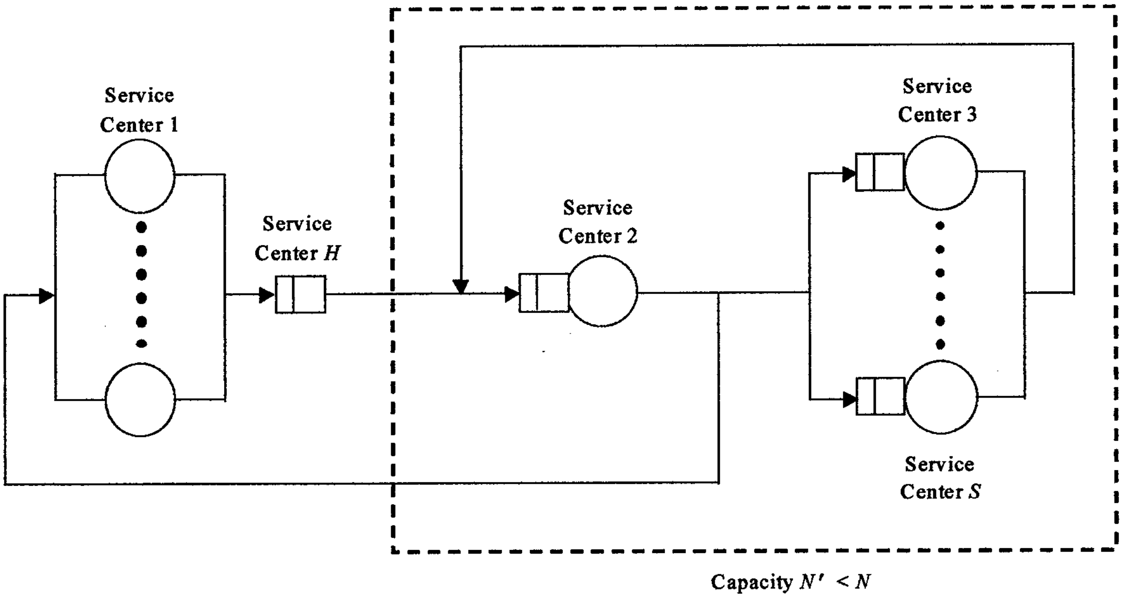

Figure 4.3 Closed queueing network Q_2

and Q_2. We estimate these steady-state performance measures by truncating a constant number of events from the beginning of each simulation replication to eliminate initial transient behavior bias. The reader should interpret all definitions of the following statistics to implicitly include these truncations.

First the system sojourn time is estimated in the following manner. Let τ be the true expected sojourn time that we wish to estimate, and let t_{ij} be the i^{th} realized sojourn time during replication j. Defining an event as the completion of service at any of the network stations, each replication will be terminated upon the completion of M events. For replication j, define m_j as the number of customer returns to station 1 at or before M events. Then calculate, T_j, the sample mean sojourn time for replication j by

$$T_j = \frac{1}{m_j} \sum_{i=1}^{m_j} t_{ij} \quad j = 1, 2, \ldots, n \tag{4.13}$$

so that τ can be estimated by

$$\hat{\tau} = \bar{T}(n) = \frac{1}{n}\sum_{j=1}^{n} T_j \tag{4.14}$$

Let v be the true steady-state value of CPU utilization, and let b_{ij} be the i^{th} service time realized at station 2 during replication j. Note that this can be assigned to any of the customers in the system. To estimate v, define e_j as the simulated time until M events occur and q_j as the number of service completions at station 2 for replication j. We then represent the CPU utilization sample mean for replication j, by U_j, where

$$U_j = \frac{1}{e_j}\sum_{i=1}^{q_j} b_{ij} \quad j = 1, 2, \ldots, n \tag{4.15}$$

so that v can be estimated as

$$\hat{v} = \bar{U}(n) = \frac{1}{n}\sum_{j=1}^{n} U_j \tag{4.16}$$

4.3.1 Internal Control Variates. As described in the chapter II, internal control variates are the random variables, or functions of them, generated within the simulation that have a known mean. Random variables meeting these criteria for models Q_1 and Q_2 include the S service times and the $S-1$ values of the routing proportions for stations $1, 3, \ldots, S$. Many researchers have explored functions of the input random variables to find robust and asymptotically stable internal control variates.

Standardized work variables are chosen for the S service time variables. Wilson and Pritzker [56] have demonstrated that standardized work variables—standardized statistics of the service time distribution at each service station—are robust and asymptotically stable for the type of queueing system studied here. The standardized work variables are defined this way. For each replication

j, let $s_{ij}(k)$ represent the i^{th} realized service time at stations $k = 1, 2, \ldots, S$. Let $a_j(k)$ be the total number of service completions at station k during replication j. Then the standardized work variables W_{kj} are given by

$$W_{kj} = \frac{1}{\sqrt{a_j(k)}} \sum_{i=1}^{a_j(k)} \frac{s_{ij}(k) - \mu_k}{\sigma_k}, \quad k = 1, 2, \ldots, S; \; j = 1, 2, \ldots, n \tag{4.17}$$

where μ_k and σ_k are given in Equations (4.11) and 4.12 respectively. Then each W_{kj} has a mean of zero and a standard deviation of one as $a_j(k) \rightarrow \infty$ [56].

Given the candidate control variates for the service time distributions, the focus is changed to the routing random variables. The choice is the standardized routing variable—a standardized statistic of the routing random variables—has been shown to significantly reduce variance based on this type of multinomial construct [10]. The standardized routing variable is developed in the following manner. An indicator variable,$I_{ij}(k)$ is defined, such that

$$I_{ij}(k) = \begin{cases} 1 & \text{if the } i^{th} \text{ station 2 departure goes to station } k \text{ for replication } j \\ 0 & \text{otherwise} \end{cases} \tag{4.18}$$

for $k = 1, 3, \ldots, S$. Here, the i^{th} departure refers to the i^{th} service completion at station 2, regardless of the customer. Then a standardized routing variable for activity k is defined as

$$R_{kj} = \sum_{i=1}^{a_j(2)} \frac{I_{ij}(k) - p_k}{\{(a_j(2))\,(1 - p_k)\,p_k\}^{1/2}}, \quad j = 1, 2, \ldots, n; \; k = 1, 3, \ldots, S \tag{4.19}$$

where $a_j(2)$ is the total number of service completions at station 2 during replication j. Each p_k is the transition probability from station 2 to station k as given in Equation (4.10). Bauer and Wilson have shown that standardized routing variables converge to a normal distribution with mean zero, as the simulation run length increases [10].

4.3.2 Analytical Control Variates. To find ACV's for models Q_1 and Q_2, consider a new model (called Q_3) that can be solved analytically. Let Q_3 have the same structure as model Q_1, except that all service time distributions are independent, identically distributed (IID) exponential random variables with mean service times equal to those of Q_1. Using a product-form algorithm such as Mean Value Analysis (MVA) [13,16,29], the steady-state expected sojourn time and CPU utilization for Q_3 can be determined exactly. As discussed in chapter II, the MVA algorithm is used to solve queueing networks that have a closed form solution. An MVA approach is chosen since it is easy to implement and reasonably fast. The algorithm yields the mean values of response time, queue length, throughput, and utilization for each service center in the network. It solves the network by first determining these values when only one customer is in the network. Using this information, the network is solved when two customers are in the system using the mean value theorem. The mean value theorem relates the response time of a service center when n customers are present to the length of the queue when $n-1$ customers are present. The algorithm is reapplied until the network is solved for the total number of customers desired.

Define $C(T)_j^{ANA}$ as the ACV for sojourn time and $C(U)_j^{ANA}$ as the ACV for CPU utilization under replication j. For each ACV calculate the sample mean service times and the realized branching proportions from station 2 for each replication and then input them into the MVA model for each simulation replication. To accomplish this, for replication j define $v_{ij}(k)$ as the i^{th} realized service time at station k. Then define V_{kj}, the average service time for replication j at station k, as

$$V_{kj} = \frac{1}{m_j(k)} \sum_{i=1}^{a_j(k)} v_{ij}(k), \quad k = 1, 2, \ldots, s; \; j = 1, 2, \ldots, n \tag{4.20}$$

where $a_j(k)$ is the number of service completions at station k during replication j. To calculate the realized routing proportions, P_{kj}, from station 2 to station k for replication j, let

$$P_{kj} = \frac{\sum_{i=1}^{a_j(k)} I_{ij}(k)}{\sum_{\substack{k=1 \\ k \neq 2}}^{S} \sum_{i=1}^{a_j(2)} I_{ij}(k)}, \quad k = 1, 3, \ldots, S; \; j = 1, 2, \ldots, n \tag{4.21}$$

where the indicator variables, $I_{ij}(k)$, are as defined in Equation (4.18). These values of V_{kj} and P_{kj} are then input into the MVA algorithm to obtain $C(T)_j^{ANA}$ and $C(U)_j^{ANA}$ for each replication j.

The next step in the ACV method is to approximate the mean of the ACV (for both system sojourn time and CPU utilization) using the Monte Carlo method described in Section 4.2.2. This method cannot be directly applied since the random variables that must generated are not independent. The routing proportions have a multinomial distribution and are therefore correlated. Even the sample mean of the realized service times are correlated to the routing proportions, since their variance is a function of the number of times customers are routed to their respective service centers.

The Monte Carlo method is modified by generating correlated random variates using the conditional distribution method described in Law and Kelton [32]. The method requires that the complete joint distribution of the random variables to be generated be known as well as the derivation of the marginal and conditional distributions. The method begins by generating a single random variate from its marginal distribution. The next random variate is generated using its marginal distribution conditioned on the realization of the first random variate. The scheme repeats this process until all random variates are generated. In this particular case, the required distributions can be derived since the routing proportions specify a multinomial distribution.

The random variates generated are the Monte Carlo routing proportions, P_{kj}^{MC}, and the Monte Carlo service times, V_{kj}^{MC}, for $k = 1, 2, \dots, S$ and $j = 1, 2, \dots, G$, where G is the number of Monte Carlo replications. To begin, construct the multinomial distribution that has the routing proportions as its parameters. The multinomial random variables, A_{kj}, are described as the number of times customers are routed from station 2 to station k for Monte Carlo replication j. The total number of routings for each Monte Carlo replication are determined based on the simulation model. Note that all customers return to station 2 following service at any other station, so for every service completion at station $k \neq 2$, there is a paired service completion at station 2. Since the total number of counted events for every simulation replication is M, the total number of routings from station 2 to the other service stations is half of that, or $M/2$. the multinomial distribution of customer routings can then be described by

$$\{A_{kj},\ k = 1, 3, \dots, S\} \sim \text{multinomial}\left([M/2]\,;\, p_1, p_3, \dots, p_S\right),\ j = 1, 2, \dots, G \tag{4.22}$$

where the p_k are as described in Equation (4.10). Also, $A_{2j} = M/2$ for all j.

The following is a scheme to generate the required random variates using the conditional distribution method of generating correlated random variates [32]. The marginal and conditional distributions are derived in the following manner. The marginal distribution of A_{kj} is binomial with parameters $[M/2]$ and $p_k (k \neq 2)$. The marginal distribution of A_{q+1}, conditional on the realization of $A_{kj} = a_{kj}$, $k = 1, 3, \dots, q$ $(q < S)$, is also binomial with parameters $[M/2] - \sum_{\substack{k=1 \\ k \neq 2}}^{q} a_{kj}$ and $p_{q+1} / \left(1 - \sum_{\substack{k=1 \\ k \neq 2}} p_k\right)$. Using the normal approximation to the binomial, generate each P_{kj}^{MC} as follows:

1. Generate $P_{1j} \sim \text{Normal}\left(p_1, \dfrac{p_1(1 - p_1)}{M/2}\right)$.
2. Let $A_{1j} = P_{1j}(M/2)$.
3. Generate $P_{3j} \sim \text{Normal}\left(\dfrac{p_3}{1 - p_1}, \dfrac{\left(\frac{p_3}{1-p_1}\right)\left(1 - \frac{p_3}{1-p_1}\right)}{M/2 - A_{1j}}\right)$.

4. Let $A_{3j} = P_{3j}\left((M/2) - A_{1j}\right)$.

5. $\vdots$

6. $\vdots$

i. Generate $P_{S-1j} \sim \text{Normal}\left(\frac{p_{S-1}}{1-\sum_{k=1}^{s-2} p_k}, \frac{\left(\frac{p_{S-1}}{1-\sum_{k=1}^{S-2} p_k}\right)\left(1-\frac{p_{S-1}}{1-\sum_{k=1}^{S-2} p_k}\right)}{\left((M/2)-\sum_{k=1}^{S-2} A_{kj}\right)}\right)$.

i+1. Let $A_{S-1j} = P_{S-1j}\left((M/2) - \sum_{k=1}^{S-2} A_{kj}\right)$.

i+2. Let $A_{Sj} = (M/2) - \sum_{k=1}^{S-1} A_{kj}$.

i+3. Let $P_{kj}^{MC} = \frac{A_{kj}}{(M/2)}$ $k = 1, 3, \ldots, S$.

Given these realizations, the distribution for each of the Monte Carlo generated mean service times is given by

$$V_{kj}^{MC} \sim \text{Normal}\left(\mu_k, \frac{\sigma_k^2}{A_{kj}}\right),\ k = 1, 2, \ldots, S \tag{4.23}$$

where $A_{2j} = M/2$ and μ_k and σ_k^2 are the known mean and variance of each service activity. Then for each Monte Carlo sample j, the generated values of the routing proportions and mean service times are input into the MVA algorithm to obtain μ_j^{MC}. The mean of the ACV can then be approximated by

$$\hat{\mu_Z} = \frac{1}{G}\sum_{j=1}^{G} \mu_j^{MC} \tag{4.24}$$

4.3.3 External Control Variates. External control variates can be obtained by creating a simulation for model Q_3. The true steady-state expected response time (τ^{EXT}) and expected CPU utilization (υ^{EXT}) are determined by the MVA algorithm. Using common random numbers, this simulation model can then be used to produce external control variates for Q_1 or Q_2. However, due to the nature of a closed network, exact synchronization of random variates becomes very difficult.

Law and Kelton [32] provide an excellent discussion of common random numbers and the problems associated with their application. For these models some common random number synchronicity can be achieved. Note that the service times at stations $1, 3, \ldots, S$ in Q_1 or Q_2 are distributed as IID Weibull random variables, and that the same service times in model Q_3 are distributed IID exponential. The inverse-transform method of random variate generation to generate the service times can then be used for both models [32]. Thus the i^{th} service time at the k^{th} service station is generated by using the same uniform [0,1] pseudo-random number for both models Q_1 (or Q_2) and Q_3. Additionally, the same uniform [0,1] pseudo-random number can be used to generate the routing random variable for the i^{th} service completion at station 2 for both models. The inability to achieve complete synchronization results from the fact that the i^{th} service time at station k will not be exactly the same for both models (albeit highly correlated). This situation eventually results in different sequences (between models) of specific customers arriving at any particular station.

To generate the external control variates for each simulation replication j, the following output statistics for simulation model Q_3 are calculated. Note, the same implicit truncation scheme is used for Q_3 as it is for Q_1. Let t_{ij}^{EXT} be the i^{th} realized sojourn time during replication j. As before, by defining an event as the completion of service at any of the network stations, each replication will be terminated upon the completion of M events. For replication j, define m_j^{EXT} as the number of customer returns to station 1 at or before M events. Then, $C(T)_j^{EXT}$, the external control variate for sojourn time for replication j, is calculated by

$$C(T)_j^{EXT} = \frac{1}{m_j^{EXT}} \sum_{i=1}^{m_j^{EXT}} t_{ij}^{EXT} \quad j = 1, 2, \ldots, n \tag{4.25}$$

For CPU utilization, let b_{ij}^{EXT} be the i^{th} service time realized at station 2 during replication j of model Q_3. Next define e_j^{EXT} as the simulated time until M events occur and q_{ij}^{EXT} as the number of service completions at station 2 for replication j. Then the external control variate for CPU utilization, $C(U)_j^{EXT}$for replication j, is calculated by

$$C(U)_j^{EXT} = \frac{1}{e_j^{EXT}} \sum_{i=1}^{q_j^{EXT}} b_{ij}^{EXT} \quad j = 1, 2, \ldots, n \tag{4.26}$$

4.4 Performance Comparison

4.4.1 Experimental Procedures. Several experiments are conducted, each varying in the number of replications and network parameters chosen. Twelve different network design points (six for Q_1 and six for Q_2) are selected. The selection of particular network settings is discussed below. At each design point, 100 experiments are conducted with the number of replications for each experiment equal to 10, and 50 experiments with the number of replications for each experiment equal to 20. For every design point, performance measures for internal, analytical, and external control variates are compared. Estimated variance, confidence interval width, coverage, and MSE values for the controlled responses are compared to the same values for the uncontrolled responses. For the internal control variates, comparisons are made for all possible combinations of input variates for both networks. The efficiency of the ACV controlled responses is also compared with that of the uncontrolled responses.

Comparisons are made using the generalized method presented by Bauer and Wilson [10]. Let μ be the expected value of the performance measurement of concern. For the $n = 10(20)$ replications of the h^{th} experiment, $h = 1, 2, \ldots, d$ $(d = 100(50))$, an estimate of μ is computed. Call the estimate $\hat{\mu}_k(l)$, where $l = 1$ and $l = 2$ denote uncontrolled and controlled estimates respectively. In a similar manner, let $\hat{\sigma}_k^2(l)$, $l = 1, 2$ denote the uncontrolled and controlled estimates of the variance of $\hat{\mu}_k(l)$. Then the average variance estimator over all d experiments for a given setting is

$$\hat{\sigma}^2(l) = \frac{1}{d} \sum_{h=1}^{d} \hat{\sigma}_h^2(l) \quad l = 1, 2 \tag{4.27}$$

The percentage change in estimated variance due to the use of a particular control variate method is then estimated by $100(\hat{\sigma}^2 - \hat{\sigma}^2(1))/\hat{\sigma}^2(1)$.

For the h^{th} experiment, the confidence interval estimate is given by

$$\hat{\Lambda}(l) = \hat{\mu}_k(l) \pm \hat{H}_k(l) \tag{4.28}$$

where $\hat{H}_k(l)$ is the estimated half-width as given in Equation (2.21) with $\alpha = 0.10$. We find the average width of the confidence interval estimator over all d experiments for a given setting as

$$2\hat{H}(l) = \frac{1}{d}\sum_{h=1}^{d} 2\hat{H}_h(l) \quad l = 1,2 \tag{4.29}$$

Then, as with the variance estimates, the percentage change in the estimated confidence interval width due to the use of a particular control variate method is estimated by $100\left(2\hat{H}(2) - 2\hat{H}(1)\right)/2\hat{H}(1)$.

An important concern for control variate performance is the amount of bias in the controlled estimate of μ. Bias is induced because β must be estimated and is generally not independent of $\bar{Y}(n)$ [30]. One related measure of bias is the estimated confidence interval coverage probability. To estimate coverage, let

$$\hat{I}_k(l) = \begin{cases} 1 & \text{if } \mu \in \hat{\Lambda}_k(l) \\ 0 & \text{otherwise} \end{cases} \tag{4.30}$$

for $l = 1,2$ and $h = 1,2,\ldots,d$. Then an estimate of the confidence interval coverage probability is given by the calculated coverage fraction for $\hat{\Lambda}_k(l)$, computed as

$$\hat{I}(l) = \frac{1}{d}\sum_{h=1}^{d} \hat{I}_k(l) \quad l = 1,2 \tag{4.31}$$

Realized coverage may not always be the most informative indicator of bias. For example, a point estimate may be very close to μ, but if the associated confidence interval is small enough, coverage may not be realized. In order to measure bias in a manner that considers this "closeness," the estimated value of the mean square error (MSE) of a point estimator is computed as

$$\hat{M}(l) = \frac{1}{m}\sum_{h=1}^{m}\left(\hat{\mu}_k(l) - \mu\right)^2 \quad l = 1, 2 \tag{4.32}$$

The true expected values of the system sojourn time and CPU utilization are estimated through 25,000 replications at each design point. With these very large samples, the associated .90 confidence intervals are sufficiently tight (less than 0.5% of estimated value in all cases) to make good benchmark estimates of coverage and MSE for comparison purposes.

Another concern is the efficiency of the ACV method. Generating the ACV for each replication and the Monte Carlo replications for approximating μ_Z consume available computer time. Depending on the amount of variance reduction achieved, a smaller confidence interval width may be achieved by simply using all available computer time to generate an uncontrolled response. The widths of confidence intervals produced by an ACV controlled response using 20 simulation replications are compared with that predicted for an uncontrolled response over an equivalent amount of time. As in section 4.2.3, we let t_{REP}, t_{ACV}, and t_{MC} represent the times required to generate a single replication of $Q_1(Q_2)$, a single ACV, and a single replication of the Monte Carlo approximation of μ_Z respectively. Then the time required to produce an ACV controlled response for 20 simulation replications is determined as

$$\mathcal{T}_{ACV} = 20\left(t_{REP} + t_{ACV}\right) + Gt_{MC} \tag{4.33}$$

The equivalent number of replications possible, if allocated all of $\mathcal{T}_{ACV}$ is allocated to replications of $Q_1(Q_2)$ only, is given by

$$n_{REP} = \left\lfloor \frac{\mathcal{T}_{ACV}}{t_{REP}} \right\rfloor \tag{4.34}$$

where $\lfloor i \rfloor$ returns the closest integer less than or equal to i. The estimated equivalent confidence interval width is

$$CIW_{EQV} = 2t_{1-\alpha/2,n_{REP}-1}\sqrt{\frac{S_Y^2}{n_{REP}}} \tag{4.35}$$

where S_Y^2 is the sample variance calculated over all 1,000 replications at the appropriate design point. CIW_{EQV} is compared to the average confidence interval width achieved by the ACV controlled response at each design point.

ACV efficiency is also examined by comparing the times required to produce an equivalent size confidence interval about an uncontrolled and ACV controlled response. The average confidence interval widths of the ACV controlled response using 20 replications are used as the base line. Then the predicted number of replications required to achieve an equivalent confidence interval width about an uncontrolled response is determined by

$$n_{EQV} = S_Y^2 \left(\frac{2t_{1-\alpha/2,n_{EQV}-1}}{CIW_{ACV}} \right)^2 \tag{4.36}$$

where S_Y^2 is the same as above and CIW_{ACV} is the average confidence interval width about the ACV controlled response. The time required to complete n_{EQV} replications of model $Q_1(Q_2)$, given by $n_{EQV}t_{REP}$, is then compared to the time required to produce an ACV controlled response of 20 replications, given by $20\,(t_{REP} + t_{ACV}) + Gt_{MC}$.

4.4.2 Network Settings. For the closed queueing network presented in Section 4.3, six different experimental design points are selected for both models Q_1 and Q_2. For all experiments, the number of service stations (S) is 6 and the number of customers (N) is 25. For model Q_2, the number of customers allowed into the constrained subnetwork (N') is 5 for all settings. To create the six design points for each model, two different transition probability matrices are applied to three sets of service time distributions. The two transition probability distributions are provided in Table 4.1 and the three service time settings are listed in Table 4.2. These settings are created to stress the queueing network at different stations to determine the effectiveness of each type of control variate under various network flow conditions. Results are obtained for each design point using replication sizes of both 10 and 20.

Table 4.1 Transition probability matrix values.

Matrix	p_1	p_3	p_4	p_5	p_6
$\mathbf{P}_1$	0.20	0.36	0.36	0.040	0.040
$\mathbf{P}_2$	0.25	0.30	0.30	0.075	0.075

In all twelve cases, replications of the simulation are terminated following the completion of 2,000 events. To remove the effect of the initial transient behavior, data from the first 500 events is ignored. In addition, the initial state of the network (number of customers at each station) is based on the expected number of customers at each station for analytical model Q_3. These expected values are determined by solving the system using Mean Value Analysis. The approximate steady-state probability that a customer is at a given station is determined by dividing the expected number of customers at a service station by the number of total customers. Then, at the start of each replication, each customer is assigned a uniform [0,1] pseudo-random number and is routed to a particular station by corresponding probabilities. G, the number of Monte Carlo replications required to approximate μ_Z, is set to equal 10,000 at all design points. The width of the 90% confidence interval about $\hat{\mu}_Z$ is approximately 10% of the width of the 90% confidence intervals

Table 4.2 Service time distribution settings.

Service Center	Distribution	α	β	Mean	Variance
Setting A					
1	Weibull	1.46824	1000.0	100.00	4795.78
2	exponential	$\sim$	$\sim$	1.00	1.00
3	Weibull	5.64760	10.0	1.39	0.08
4	Weibull	5.64760	10.0	1.39	0.08
5	Weibull	2.61249	1000.0	12.50	26.44
6	Weibull	2.61249	1000.0	12.50	26.44
Setting B					
1	Weibull	1.46824	1000.0	100.00	4795.78
2	exponential	$\sim$	$\sim$	1.00	1.00
3	Weibull	1.50438	10.0	4.17	7.97
4	Weibull	5.64760	10.0	1.39	0.08
5	Weibull	2.61249	1000.0	12.50	26.44
6	Weibull	2.61249	1000.0	12.50	26.44
Setting C					
1	Weibull	1.46824	1000.0	100.00	4795.78
2	exponential	$\sim$	$\sim$	1.00	1.00
3	Weibull	5.64760	10.0	1.39	0.08
4	Weibull	5.64760	10.0	1.39	0.08
5	Weibull	2.06810	1000.0	25.00	160.80
6	Weibull	2.61249	1000.0	12.50	26.44

about the ACV controlled responses at all design points when $G = 10,000$. This value of G also worked well in terms of ACV efficiency.

4.4.3 Results. ACV's provide significant confidence interval reduction on estimates for both system sojourn time and CPU utilization for the closed queueing network. Across the range of all experiments, ACV performance is typically similar to that of external and internal methods. Confidence interval width reductions, as a percentage of the uncontrolled estimated confidence interval, are provided in Tables 4.3 and 4.4. Results are included for ACV's, external control variates, and internal control variates. The internal case represents the combination of standardized work variables and standardized routing variables that produces the greatest reduction in confidence interval width. Due to the similarity in variance reduction achieved, only results for 20 replications are provided.

Table 4.3 Confidence interval width reduction (System sojourn time)

System sojourn time					
			Confidence interval width reduction (%)		
Model	Service time setting	Transition probability matrix	Analytical control variate	External control variate	Internal control variates
Q_1	A	$\mathbf{P}_1$	53.7	52.5	56.9
		$\mathbf{P}_2$	46.0	50.4	45.0
	B	$\mathbf{P}_1$	47.5	64.2	66.4
		$\mathbf{P}_2$	49.9	57.2	52.4
	C	$\mathbf{P}_1$	28.5	55.7	43.6
		$\mathbf{P}_2$	22.5	50.8	64.3
Q_2	A	$\mathbf{P}_1$	58.0	40.0	49.2
		$\mathbf{P}_2$	54.0	32.3	46.0
	B	$\mathbf{P}_1$	62.3	39.8	68.0
		$\mathbf{P}_2$	59.0	22.4	46.8
	C	$\mathbf{P}_1$	51.9	24.4	49.7
		$\mathbf{P}_2$	58.2	10.6	68.6

Of particular interest is the performance of external and analytical methods for model Q_2. The external method fails to provide the same level of confidence interval width reduction (particularly for CPU utilization) for model Q_2 as for Q_1. Recall that only five customers at a time are allocated a portion of main memory, whereas model Q_1 has an unconstrained subnetwork and an unlimited number of customers may access the CPU and storage devices. The variance in external control variate performance is probably due to this dissimilarity in model structure. For model Q_2, the common random numbers lose more synchronization and the system responses are not as highly correlated. ACV's, on the other hand, continue to perform at about the same level for Q_2 as for Q_1. Although the underlying analytical model is also unconstrained, the model relies only on the mean responses of system parameters from Q_2. Therefore, the same conditions that cause longer sojourn times or greater CPU utilization levels in model Q_2 will do so in the analytical model. Hence, correlation from replication to replication is maintained and ACV's perform well for Q_2.

Realized coverage and estimated MSE estimates for uncontrolled and controlled responses are enumerated in Tables 4.5 and 4.6. Nominal coverage is 90%. Due to the similarity in the results,

Table 4.4 Confidence interval width reduction. (CPU utilization)

CPU utilization					
Model	Service time setting	Transition probability matrix	Confidence interval width reduction (%)		
			Analytical control variate	External control variate	Internal control variates
Q_1	A	$\mathbf{P}_1$	55.4	46.8	47.1
		$\mathbf{P}_2$	60.5	50.1	54.3
	B	$\mathbf{P}_1$	70.6	63.3	81.1
		$\mathbf{P}_2$	57.9	52.0	47.8
	C	$\mathbf{P}_1$	42.7	55.8	34.8
		$\mathbf{P}_2$	45.0	51.6	83.6
Q_2	A	$\mathbf{P}_1$	38.3	16.2	34.9
		$\mathbf{P}_2$	43.7	11.0	30.9
	B	$\mathbf{P}_1$	65.4	16.6	65.5
		$\mathbf{P}_2$	33.0	3.6	32.4
	C	$\mathbf{P}_1$	44.9	10.5	42.3
		$\mathbf{P}_2$	69.4	9.7	71.3

only the statistics for 20 replications are provided. Internal control variate results correspond to the same internal control variate combinations reported above for confidence interval width reduction.

The approximation of μ using the Monte Carlo technique has not induced any detectable bias in the analytically controlled estimates of sojourn time and CPU utilization. Both realized coverage percentages and estimated MSE's are similar for all three types of control variates, with no indications of any significant bias. Additionally, the estimated MSE's for all controlled responses are smaller than those of the uncontrolled estimates.

Figures 4.4 and 4.5 illustrate the performance of the ACV compared to the uncontrolled response and the internal and external controlled responses. The figures also depict the associated confidence intervals for 50 experiments at a particular network setting and 20 replications per experiment. For system sojourn time, model Q1 with service time setting B, transition probability matrix P2, and 20 replications (50 experiments) is presented. The 50 diamond symbols in each figure represent the 50 point estimates. The bracketed lines above and below each diamond represent the width of the estimated confidence interval for the associated 20 replication design point. For

Table 4.5 Realized coverage (nominal = 90%) and estimated MSE. (System sojourn time)

System sojourn time						
Model	Service time setting	Transition probability matrix	Coverage percentage (Estimated MSE)			
			Uncontrolled response	Analytical control variate	External control variate	Internal control variates
Q_1	A	$\mathbf{P}_1$	92 (3.15)	96 (0.61)	90 (0.71)	86 (0.63)
		$\mathbf{P}_2$	86 (2.98)	82 (0.69)	90 (0.51)	84 (0.84)
	B	$\mathbf{P}_1$	86 (22.53)	88 (5.16)	94 (1.99)	88 (2.07)
		$\mathbf{P}_2$	84 (5.24)	94 (0.84)	92 (0.59)	88 (1.07)
	C	$\mathbf{P}_1$	94 (5.67)	86 (3.68)	82 (1.47)	88 (2.48)
		$\mathbf{P}_2$	88 (38.64)	88 (19.53)	88 (8.24)	90 (3.94)
Q_2	A	$\mathbf{P}_1$	90 (5.83)	98 (0.72)	92 (1.39)	80 (1.62)
		$\mathbf{P}_2$	88 (6.39)	94 (1.07)	94 (1.73)	90 (1.41)
	B	$\mathbf{P}_1$	90 (25.90)	86 (3.81)	90 (7.76)	94 (1.68)
		$\mathbf{P}_2$	94 (8.58)	96 (1.43)	96 (4.58)	88 (2.36)
	C	$\mathbf{P}_1$	98 (15.61)	88 (3.42)	96 (8.03)	96 (2.62)
		$\mathbf{P}_2$	88 (54.50)	90 (8.13)	86 (35.26)	88 (4.26)

reference, the estimated mean found using 25,000 replications is represented by the horizontal line across each figure. For CPU utilization, model Q2 with service time setting C, probability transition matrix P2, and 20 replications is used to create Figure 4.5.

The figures not only illustrate the confidence interval width reduction achieved when using control variate methods, but also illustrate the improved accuracy of the point estimates (or consistently lower values for estimated MSE) for the controlled responses when compared to the uncontrolled responses. Additionally, bias does not appear to be a problem with any of the control variates illustrated, even for the ACV's using Monte Carlo approximations of $\hat{\mu}_Z$.

ACV efficiency comparisons are provided in Tables 4.7 and 4.8 for both time equivalent and confidence width equivalent results. All comparisons are based on ACV confidence intervals produced using 20 replications of the simulation model. All computer generation times are calculated using our results on a Sun SPARC 2 workstation. The computer times are $t_{REP} = 1.8sec$, $t_{ACV} = 0.01sec$, *nd* $t_{MC} = 0.003sec$, with $\mathcal{T}_{ACV} = 66.2sec$ and $n_{REP} = 36$ replications for the time equivalent comparisons. The comparison ratios indicate the relative efficiency of the ACV by comparing

Table 4.6 Realized coverage (nominal = 90%) and estimated MSE. (CPU utilization)

System sojourn time						
Model	Service time setting	Transition probability matrix	Coverage percentage (Estimated MSE)			
			Uncontrolled response	Analytical control variate	External control variate	Internal control variates
Q_1	A	$\mathbf{P}_1$	90 (6.44)	86 (1.83)	92 (2.36)	88 (2.09)
		$\mathbf{P}_2$	90 (9.08)	88 (1.47)	84 (2.88)	88 (2.59)
	B	$\mathbf{P}_1$	88 (10.72)	82 (1.23)	80 (1.77)	92 (0.48)
		$\mathbf{P}_2$	92 (5.82)	98 (0.82)	92 (1.26)	90 (1.49)
	C	$\mathbf{P}_1$	92 (18.10)	84 (6.82)	82 (5.88)	88 (7.33)
		$\mathbf{P}_2$	86 (26.70)	90 (5.75)	80 (8.84)	86 (0.55)
Q_2	A	$\mathbf{P}_1$	90 (4.82)	90 (2.24)	94 (3.18)	86 (2.72)
		$\mathbf{P}_2$	84 (11.10)	86 (2.64)	84 (9.02)	86 (3.84)
	B	$\mathbf{P}_1$	92 (7.35)	94 (0.63)	94 (4.11)	88 (0.70)
		$\mathbf{P}_2$	88 (5.25)	96 (1.68)	88 (4.81)	84 (2.37)
	C	$\mathbf{P}_1$	90 (18.10)	94 (4.24)	88 (14.70)	96 (3.66)
		$\mathbf{P}_2$	90 (18.60)	92 (1.99)	86 (13.20)	92 (1.77)

the time required to produce a 20 replication ACV controlled response to the time equivalent and confidence interval width equivalent times of the uncontrolled response. Values greater than 1 indicate that the ACV controlled response is more efficient than the uncontrolled response; values less than 1 indicate that the equivalent uncontrolled response is more efficient than the ACV controlled response.

The results indicate that the ACV method is more efficient for all performance measures at all design points except for one. Even in that case, the comparison ratios are nearly equal to 1. Given that the coverage and MSE estimates for the ACV method indicate no detectable bias the 10% guideline for the $\hat{\mu}_Z$ confidence interval width to that of the ACV controlled response confidence interval width appears to be appropriate. Further, this choice of G has provided an ACV controlled response that is efficient as well.

Although there are 12 different design points, only 6 different Monte Carlo approximations of μ_Z had to be calculated since the same analytical model is used for both Q_1 and Q_2. In order to be fair the efficiency comparisons are made as if $\hat{\mu}_Z$ is calculated at every design point, when

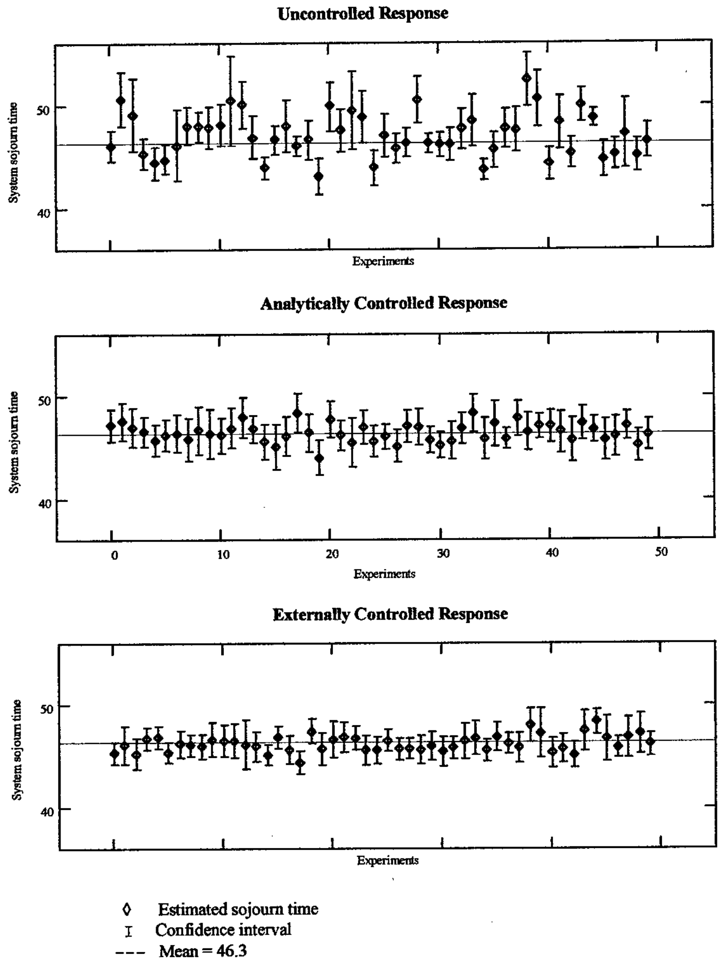

Figure 4.4 Experimental results with estimated confidence interval. Design point: Q_1, service time setting B, transition matrix $\mathbf{P}_2$.

in actuality the "true" efficiency is twice that reported. Depending on the type of experimental design required for a particular simulation study, this same situation can occur, further increasing the efficiency of the ACV controlled response.

4.5 Conclusion

This chapter demonstrates that a hybrid type of control variate, called an ACV, can effectively reduce point estimate variance from replicative simulation studies while avoiding some of the technical difficulties of internal and external control variates. In terms of confidence interval

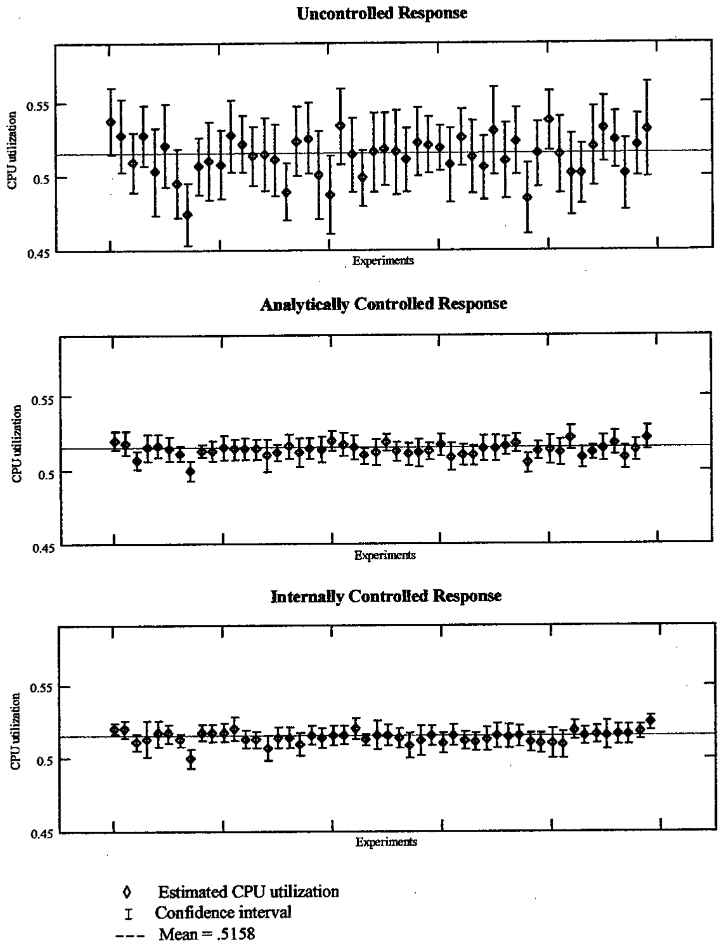

Figure 4.5 Experimental results with estimated confidence interval. Design point: Q_2, service time setting C, transition matrix $\mathbf{P}_2$.

width reduction, the experimental results indicate that ACV's are quite successful for the networks studied. This supports the findings of previous researchers.

Previous researchers also reported unacceptable levels of bias for point estimates found using ACV's. This bias is caused by the use of inaccurate values for the mean of the analytical model when given inputs from the simulation model. This research presents a simple Monte Carlo method for approximating the mean of the analytical model that eliminates detectable bias.

The Monte Carlo method should be general enough to be used for many discrete event simulation model where the distribution of the observed sample means of the simulation input random

Table 4.7 Efficiency comparisons. (System sojourn time)

System sojourn time							
				Time Equivalent		CIW Equivalent	
Model	Service time setting	Transition probability matrix	ACV CIW	UNC CIW	Efficiency ratio	UNC Rep.'s	Efficiency ratio
Q_1	A	$\mathbf{P}_1$	2.66	4.26	1.60	89	2.43
		$\mathbf{P}_2$	2.56	3.55	1.39	68	1.85
	B	$\mathbf{P}_1$	7.40	10.46	1.41	70	1.91
		$\mathbf{P}_2$	3.36	4.99	1.49	78	2.13
	C	$\mathbf{P}_1$	5.94	6.18	1.04	39	1.06
		$\mathbf{P}_2$	14.33	13.72	0.96	34	0.93
Q_2	A	$\mathbf{P}_1$	3.27	5.77	1.76	108	2.95
		$\mathbf{P}_2$	3.59	5.86	1.63	93	2.54
	B	$\mathbf{P}_1$	5.91	11.62	1.97	134	3.65
		$\mathbf{P}_2$	4.23	7.61	1.80	113	3.08
	C	$\mathbf{P}_1$	6.86	10.55	1.54	83	2.26
		$\mathbf{P}_2$	8.77	15.65	1.78	111	3.03

variables is known either exactly or approximately. The use of the conditional distribution method for generating correlated random variates given a multinomial distribution should be applicable to almost any discrete event simulation model that contains probabilistic branching. However, for many simulation models the distribution of the observed sample means of some or all of the simulation input random variables may not be known either exactly or approximately. In those cases a re-sampling (bootstrap) method or a combination re-sampling and Monte Carlo method could be used to approximate the mean of the analytical model. Further research into these types of models to determine the accuracy and efficiency of re-sampling or a combination re-sampling and Monte Carlo method could increase the number of simulation models where an ACV could be used. Exactly this type of research is conducted in the next chapter.

Another possible problem for the Monte Carlo method could occur if very few observations are obtained for a particular input over the course of a replication. Under these conditions, the central limit theorem effect may not be powerful enough and a normal distribution assumption may be inappropriate. This may or may not be a problem though. A small number of observations

Table 4.8 Efficiency comparisons. (CPU utilization)

System sojourn time							
Model	Service time setting	Transition probability matrix	ACV CIW	Time Equivalent		CIW Equivalent	
				UNC CIW	Efficiency ratio	UNC Rep.'s	Efficiency ratio
Q_1	A	$\mathbf{P}_1$	0.0135	0.0221	1.64	94	2.56
		$\mathbf{P}_2$	0.0128	0.0238	1.87	121	3.30
	B	$\mathbf{P}_1$	0.0101	0.0252	2.51	216	5.89
		$\mathbf{P}_2$	0.0114	0.0201	1.76	108	2.95
	C	$\mathbf{P}_1$	0.0247	0.0319	1.29	59	1.61
		$\mathbf{P}_2$	0.0270	0.0364	1.35	64	1.75
Q_2	A	$\mathbf{P}_1$	0.0160	0.0191	1.19	51	1.39
		$\mathbf{P}_2$	0.0172	0.0227	1.32	62	1.69
	B	$\mathbf{P}_1$	0.0092	0.0197	2.14	159	4.34
		$\mathbf{P}_2$	0.0154	0.0170	1.10	44	1.20
	C	$\mathbf{P}_1$	0.0261	0.0348	1.33	63	1.72
		$\mathbf{P}_2$	0.0143	0.0343	2.41	200	5.45

for a particular input could indicate that the activity it represents has little effect on the overall performance of the system under study. Hence, an inappropriate approximation of its distribution could have little effect on the accuracy of our approximation of μ_Z.

Although the use of a Monte Carlo approach to approximating the mean of the analytical model requires some effort and computer time, the extra effort can pay off given sufficient variance reduction is achieved by the ACV. The required programming is very simple and the computer time required to generate Monte Carlo observations is very small. The results of this chapter indicate that the Monte Carlo method can be an efficient means of applying an ACV.

V. ACV Monte Carlo Method with Incomplete Distribution Knowledge

5.1 Overview

In the previous chapter, it was shown that the Monte Carlo method for approximating the mean of the ACV is an effective means of resolving the ACV bias problem. In order to use the Monte Carlo method, the means and variances of each of the random variables used as inputs to the analytical model must be known. In addition, the probability distribution, including the dependence relationships, of all the input variables must be known either exactly or approximately. In general, these conditions will not always be met. Depending on the underlying system and the way a simulation model of the system is constructed, the means, variances, or dependence structure of some or all of the inputs to the analytical model may not be known. In fact, a simulation model, vice an analytical model, may be constructed for exactly those reasons. Rule based routings of customers or resources is just one example. In this chapter, methods for approximating the ACV mean when some or all of these conditions are not met are explored.

The chapter begins with descriptions of different methods of generating random vectors. Non-parametric methods based on an observed random sample are described first, followed by a discussion on parametric methods of generating random vectors. These methods are tested on a simulation model based on the AMC BRACE airfield simulation model. This is an excellent model to use, since some of the sample means produced by the model, necessary to generate an ACV, have unknown means and variances. Both the simulation model and the analytical model used to produce an ACV are described in the following section. Experimental methods and results are presented in the final sections of this chapter.

5.2 Non-parametric Approximation Methods

The Monte Carlo method described in Chapter IV approximates μ_Z by generating random variate vectors based on a parametric distribution approximation of the input to the analytical model, $\bar{\mathbf{Y}}_A^S$. There exist non-parametric methods that don't require any assumptions about the underlying distribution to generate additional random vectors based on the observed data. These methods rely on some form of re-sampling of the observed data. The two re-sampling methods explored are the bootstrap and SIMDAT methods.

5.2.1 Bootstrap. One approach that doesn't require any assumptions about the distribution of the input to the analytical model or any explicit knowledge of the correlation structure is the bootstrap re-sampling technique. Efron first described the bootstrap in an attempt to better understand the jackknife estimator of the standard deviation of a distribution [20] and has since been applied to numerous statistical problems. The bootstrap is described in the following manner [21]. Consider a random variable $R(\mathbf{X}, F)$ where $\mathbf{X} = (X_1, X_2, \ldots, X_n)'$ is a random sample that is IID distributed from distribution F. Then the bootstrap estimate of $E[R]$ designated by $E^*[R^*]$ is found in the following manner:

1. Form the non-parametric MLE of F (also referred to as the empirical distribution)

$$\hat{F}: \text{ mass } 1/n \text{ at } x_i, \quad i = 1, 2, \ldots, n \tag{5.1}$$

2. Draw a "bootstrap sample" (with replacement) from $\hat{F}$

$$\mathbf{X}^* = X_1^*, X_2^*, \ldots, X_n^* \overset{IID}{\sim} \hat{F} \tag{5.2}$$

   and calculate $R^* = R(\mathbf{X}^*, \hat{F})$.

3. Independently repeat step 2 a large number of B times, obtaining "bootstrap replications": $R_1^*, R_2^*, \ldots, R_B^*$ and calculate

$$E^*[R^*] = \frac{1}{B} \sum_{b=1}^{B} R_b^* \tag{5.3}$$

Since R is any random variable based on a sample from a parent distribution, the bootstrap technique could be used to estimate the mean of the ACV given by $E[f^A(\phi^A, \bar{\mathbf{Y}}_A^S)] = \mu_Z$. If $Cov\left[\bar{\mathbf{Y}}_{A(i)}^S, \bar{\mathbf{Y}}_{A(j)}^S\right] = 0$ for $i = 1, 2, \ldots, q$; $j = 1, 2, \ldots, q$; $i \neq j$ the joint empirical distribution of the observed values over n replications of $\bar{\mathbf{Y}}_{A(j)}^S = (\bar{Y}_{A(1j)}^S, \bar{Y}_{A(2j)}^S, \ldots, \bar{Y}_{A(qj)}^S)'$ for $j = 1, 2, \ldots, n$, is given by

$$\hat{F}: \ \left(\text{mass } 1/n \text{ at } \bar{Y}_{A(1j)}^S, \text{ mass } 1/n \text{ at } \bar{Y}_{2j}, \ldots, \text{mass } 1/n \text{ at } \bar{Y}_{A(qj)}^S\right) \tag{5.4}$$

for $j = 1, 2, \ldots, n$. Hence there are q^n different vectors that can be sampled. A bootstrap estimate of $E[f^A(\phi^A, \bar{\mathbf{Y}}_A^S)]$ can be found by first drawing a bootstrap sample from $\hat{F}$

$$\bar{\mathbf{Y}}_A^{S*} = \bar{\mathbf{Y}}_{A(1)}^{S*}, \bar{\mathbf{Y}}_{A(2)}^{S*}, \ldots, \bar{\mathbf{Y}}_{A(n)}^{S*} \overset{IID}{\sim} \hat{F} \tag{5.5}$$

with $\bar{\mathbf{Y}}_{A(j)}^{S*} = (\bar{Y}_{A(1j)}^{S*}, \bar{Y}_{A(2j)}^{S*}, \ldots, \bar{Y}_{A(qj)}^{S*})'$ for $j = 1, 2, \ldots, n$ and calculate f^{A*} by

$$f^{A*} = \frac{1}{n} \sum_{j=1}^{n} f^A(\phi^A, \bar{\mathbf{Y}}_{A(j)}^{S*}) \tag{5.6}$$

Then $E[f^A(\phi^A, \bar{\mathbf{Y}}_A^S)]$ is estimated by repeating the steps described by Equations (5.5) and (5.6) B times and finding

$$E^*[f^{A*}] = \frac{1}{B} \sum_{b=1}^{B} f_b^{A*} \tag{5.7}$$

However, if the elements of $\bar{\mathbf{Y}}_{A(i)}^S$ are not independent, the empirical distribution is defined by

$$\hat{F}: \ \text{mass } 1/n \text{ at} \bar{\mathbf{Y}}_{A(j)}^S \ j = 1, 2, \ldots, n \tag{5.8}$$

Any resulting bootstrap sample will re-sample only the original n data points, offering little gain in accuracy over a sample mean of the n observed values of Z. Therefore, the bootstrap is best suited for situations where the random variates are independent, or nearly so.

5.2.2 SIMDAT. Taylor and Thompson [50] developed an algorithm for generating random vectors based on the observed values of a multivariate random vector $\mathbf{X}$. The method is referred to as the SIMDAT method. The algorithm generates "pseudo-data points" that behave as though they come from the underlying distribution of $\mathbf{X}$ without knowing or estimating the underlying distribution. Instead, the observations are combined using stochastic multipliers to generate the pseudo-observations.

The SIMDAT algorithm is described in the following manner [52]. Assume the goal is to generate pseudo-random data points from the underlying, unknown, distribution of a random sample $\{\mathbf{X}_j\}_{j=1}^n$ where $\mathbf{X}_j = (x_{1j}, x_{2j}, \ldots, x_{kj})'$. The first step is to standardize the sample points so that the marginal sample variance of each random variable is roughly the same. For some given integer m, find the $m-1$ nearest neighbors of each of the n different random samples and store the results in an array of size $n \times (m-1)$.

The goal is to generate a pseudo-sample of size N. Unlike the bootstrap, there is no need for N to equal n. Now select one of the n observed data points with probability $1/n$ and retrieve it's $m-1$ nearest neighbors in order to calculate the mean of the resulting m data points by

$$\bar{\mathbf{X}} = \left(\frac{1}{m}\sum_{i=1}^{m} x_{1i}, \frac{1}{m}\sum_{i=1}^{m} x_{2i}, \ldots, \frac{1}{m}\sum_{i=1}^{m} x_{ki} \right)' \tag{5.9}$$

Next, code the selected m data points about $\bar{\mathbf{X}}$ as

$$\{\mathbf{X}_j^C\} = \{\mathbf{X}_j - \bar{\mathbf{X}}\}_{j=1}^m \tag{5.10}$$

Although these computations of the sample means and coded variables are presented as part of the simulation process, these computations need only be done once before the simulation process begins. As with the $m-1$ nearest data points, the resulting $\mathbf{X}_j^C$ and $\bar{\mathbf{X}}$ values can be stored in an array of size $n \times (m+1)$ for later recall.

The next step is to generate m random variables, $u_1, u_2, \ldots, u_m$, from the univariate uniform distribution defined by

$$U\left[\frac{1}{m} - \sqrt{\frac{3(m-1)}{m^2}}, \frac{1}{m} + \sqrt{\frac{3(m-1)}{m^2}}\right] \tag{5.11}$$

These random variables are then used to generate a centered pseudo-data point $\mathbf{X}^C$ by

$$\mathbf{X}^C = \sum_{l=1}^{m} u_l (x_{1l}^C, x_{2l}^C, \ldots, x_{kl}^C)' \tag{5.12}$$

The pseudo-data point $\mathbf{X}_p$ is obtained by adding back $\bar{\mathbf{X}}$ to the centered pseudo-data point

$$\mathbf{X}_p = \mathbf{X}^C + \bar{\mathbf{X}} \tag{5.13}$$

These procedures are then repeated N times to generate the required pseudo-data points.

The algorithm is motivated in the following manner [50]. Consider a sampled vector $\mathbf{X}_1$ and its $m-1$ nearest neighbors

$$\{\mathbf{X}_l\}_{l=1}^m = \left\{(x_{1l}, x_{2l}, \ldots, x_{kl})'\right\}_{l=1}^m \tag{5.14}$$

Assume that the observed data points are from a truncated distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ. Further let $\{u_l\}_{l=1}^m$ be an independent sample from the uniform distribution defined in (5.11) above. Therefore $E[u_l] = 1/m$, $Var[u_l] = (m-1)/m^2$, and $Cov[u_i, u_j] = 0$, for $i \neq j$.

By forming the linear transformation

$$\mathbf{Z} = \sum_{l=1}^{m} u_l \mathbf{X}_l \tag{5.15}$$

the r^{th} component of Z is $z_r = u_1 x_{r1} + u_2 x_{r2} + \cdots + u_m x_{rm}$ resulting in the following relationships:

$$E[z_r] = \mu_r \tag{5.16}$$

$$Var[z_r] = \sigma_r^2 + \{(m-1)/m\}\, \mu_r^2 \tag{5.17}$$

$$Cov[z_r, z_s] = \sigma_{rs} + \{(m-1)/m\}\, \mu_r \mu_s \tag{5.18}$$

Observe that the linear transformation results in $\mathbf{Z}$'s that are uncorrelated. If the mean vector of $\mathbf{X}$ is $\boldsymbol{\mu} = (0, 0, \ldots, 0)'$, the mean vector and covariance matrix of $\mathbf{Z}$ is identical to that of $\mathbf{X}$ since $E[z_r] = 0$, $Var[z_r] = \sigma_r^2$, and $Cov[z_r, z_s] = \sigma_{rs}$. For the SIMDAT algorithm, the translation to the local mean of the nearest neighbor cloud will not achieve these results exactly. However it is argued that the SIMDAT algorithm generates points having very nearly the same mean and covariance structure as the underlying distribution of the points in the nearest neighbor cloud [50, 52].

The selection of the appropriate value of m is the next problem to consider. For m moderately large, by the central limit theorem, SIMDAT approximately samples from n normal distributions with mean and covariance matrices corresponding to those of the n, m nearest neighbor clouds [52]. There are rules for the consistency of the non-parametric density estimator that correspond to SIMDAT, however the formulas require values that are not normally available [52]. However, the goal is to generate new data points that resemble those observed, not construct a density estimator. Note, that if $m = 1$ the resulting estimator is the bootstrap, and if $m = n$, the samples are from an approximate normal distribution with mean vector and covariance matrix as estimated by the observed data. Guidelines based on experience are to choose $m \approx .02n$ for data sizes of approximately 1,000 or larger [52]. For smaller samples, $m \approx .05n$ has worked well [52].

The SIMDAT method can be applied for approximating the mean of the ACV regardless of any knowledge of the moments and/or dependence structure of $\bar{\mathbf{x}}$. Experiments using SIMDAT are performed with the results reported later in this chapter.

5.3 Parametric Methods

When generating an ACV, the inputs to the analytical model are output sample means of the simulation model. In chapter IV, this fact was used by the Monte Carlo method to assume a normal distribution for each of the elements of $\bar{\mathbf{x}}$. For the purposes of this section, that assumption is still valid. The difference is that some, or all, of the parameters of the assumed multivariate normal distribution are unknown. A parametric method of generating new data points could be constructed by sampling from a multivariate normal distribution where the unknown (or all) the parameters are estimated from the observed data.

Assume that a random sample $\{\mathbf{X}_j\}_{j=1}^{n}$ where $\mathbf{X}_j = (x_{1j}, x_{2j}, \dots, x_{kj})'$ is observed having been generated from the multivariate distribution given by

$$\mathbf{X}_j \sim Normal\,(\boldsymbol{\mu}, \Sigma)\;\; j = 1, 2, \dots, k \tag{5.19}$$

where

$$\boldsymbol{\mu} = (\mu_1, \mu_2, \dots, \mu_k)' \tag{5.20}$$

and

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1k} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2k} \\ \vdots & \vdots & & \vdots \\ \sigma_{k1} & \sigma_{k2} & \cdots & \sigma_k^2 \end{bmatrix} \tag{5.21}$$

Further assume that all parameters (elements of $\boldsymbol{\mu}$ and Σ) of the distribution are unknown. Then the mean parameters can be estimated by $\hat{\boldsymbol{\mu}}$ where

$$\hat{\boldsymbol{\mu}} = \bar{\mathbf{X}} = n^{-1}\mathbf{1}'\mathbf{X} \tag{5.22}$$

where $\mathbf{1}$ is a $n \times 1$ vector of ones. The variance and covariance parameters are estimated by $\hat{\Sigma}$ given by

$$\hat{\Sigma} = \mathbf{C} = (n-1)^{-1}\left[\mathbf{X}'\mathbf{X} - n^{-1}(\mathbf{X}'\mathbf{1})(\mathbf{1}'\mathbf{X})\right] \tag{5.23}$$

Then N random vectors can be generated by sampling N times from a multivariate normal distribution with parameters given by $\bar{\mathbf{X}}$ and $\mathbf{C}$ using an appropriate random variate generation scheme.

One simple scheme for generating multivariate normal random vectors is provided in Law and Kelton [32] attributed to Scheuer and Stoller [46]. Assume the goal is to generate random vectors from a multivariate normal distribution of dimension k with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ. Since Σ is symmetric and positive definite, it can be factored uniquely as $\Sigma = \mathbf{C}_{lt}\mathbf{C}_{lt}'$ where the $n \times n$ matrix $\mathbf{C}_{lt}$ is lower triangular. This is referred to as *Cholesky factorization.* An algorithm for generating the required multivariate normal random vector $\mathbf{X}$ is given by:

1. Generate the random vector $\mathbf{Z} = (Z_1, Z_2, \ldots, Z_k)'$ where each Z_i is distributed as an i.i.d. Normal$(0,1)$ random variable.

2. Let $\mathbf{X} = \hat{\boldsymbol{\mu}} + \mathbf{C}_{lt}\mathbf{Z}$.

When all parameters are known, this approach is another way of performing the Monte Carlo method described in Chapter IV. For the assumption we made here (all the parameters are unknown) estimated parameters can be used in the above scheme.

As mentioned in the introduction to this chapter, the situation is that all or *some* of the parameters of the input random vector to the ACV are unknown. In the above development, the assumption is that all parameters are unknown. To account for the situation when some of the parameters are known, the above scheme can be modified by simply replacing the estimated parameter(s) with the appropriate known parameter(s). For the unknown parameters, the estimated parameters are still used.

By assuming that $\bar{\mathbf{Y}}^S_{A(j)}$ is approximately distributed by a multivariate normal distribution, the parametric scheme described above can be used to approximate the mean of the ACV when some (or all) of the parameters of $\bar{\mathbf{Y}}^S_{A(j)}$ are unknown. Experiments using the parametric scheme are performed with the results reported later in this chapter.

5.4 Combined Methods

Another approach to generating the random vectors necessary to approximate the mean of the ACV is to combine the non-parametric and parametric methods described above. In other words, for those random variates that can be generated by a parametric approach, do so. The other random variates could then be generated using one of the other non-parametric methods. Two combined methods are described below.

To demonstrate the combination methods, consider the following. Let $\{\mathbf{X}_j\}_{j=1}^n$ where $\mathbf{X}_j = (x_{1j}, x_{2j}, \ldots, x_{kj})'$ be a random sample generated from a multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ as described above. Assume that for some of the elements of $\mathbf{X}$ the mean and variance is known and for the remaining elements, these expectations are unknown. Without loss of generality, for a k dimensional vector, let the first $l < k$ components of $\mathbf{X}$ have known means and variances and the remaining $k-l$ components have unknown means and variances. Further assume that the first l random variates are independent of the last $k-l$ random variates.

The first combination method considered combines the parametric method of section 5.3 with the bootstrap method described in Section 5.2.1. For this method, a further assumption is made, namely that the last $k-l$ random variates are also independent of each other. Simply, to generate a new random vector, the first l random variates are generated using a parametric method with known parameters. The remaining components are re-sampled from the observed data using the bootstrap method.

The other combination method joins the parametric method of Section 5.3 with the SIMDAT method of Section 5.2.2. As with the other combination method, the first l random variables are generated with a parametric method using the known parameters. The other $k-l$ variables are generated using the SIMDAT method. In this case, the last $k-l$ variables are considered to be a random vector of dimension $k-l$ for purposes of the SIMDAT algorithm.

Both combination methods are used to estimate the mean of the ACV later with results provided later in this chapter.

5.5 Airfield Operation Example

5.5.1 Overview. A discrete event simulation model of an AMC airfield is used to explore the effectiveness of approximating the mean of an ACV for each of the random vector generation methods described above. An MVA nested fork-join analytical model is also described for the purposes of generating the ACV. The simulation model is written using the SLAM II simulation language and is based on AMC's BRACE simulation model [1]. The simulation model is referred to as *Pseudo-BRACE*. This model is used in place of BRACE since it operates faster and is easier to modify so is therefore better suited for the current stage of exploratory research. The section is organized in the following manner. The simulation model and the performance measures of interests are described first, followed by a description of the analytical model and how it measures the same performance measures.

5.5.2 Simulation Model. Pseudo-BRACE models the operation of an USAF airlift airfield. The major activities of such an airfield are simulated within the model. These activities include landing, taxiing, parking, refueling, scheduled and unscheduled maintenance, and cargo upload. Each of these activities are discussed below with the aircraft arrival and parking process discussed first.

Simulated aircraft arrive at the airfield according to a Poisson process. Several different types of aircraft are simulated. A portion of those aircraft is designated to carry hazardous cargo. The airfield has a finite number of spots on its ramp, equal to P, for parking the aircraft with only some of the spots considered safe for hazardous cargo. In addition, some (not all) of the non-hazardous cargo spots are equipped with fuel hydrant refueling pits. If the arriving aircraft is designated for hazardous cargo and there is an empty hazardous cargo parking spot, the aircraft will enter the queue of aircraft waiting to use the runway. If there is no hazardous cargo spot available, the simulated aircraft will wait for two hours for a hazardous cargo spot to open up. If no spot becomes available in that time, the aircraft will leave the system; *divert* to another airfield. Aircraft not designated for hazardous cargo will enter the runway queue when any parking spot is available. The aircraft will be parked using the following preferences. The aircraft will park at a spot with a fuel hydrant spot first, if one is available. If not, the aircraft will park in a non-hazardous parking spot that doesn't have a fuel hydrant pit. Finally, the aircraft will be parked at a hazardous parking spot. Again, if the aircraft has not entered the runway queue within two hours of arrival it will divert. Only one aircraft at a time is allowed to use the runway. Landing and taxiing are simulated using a fixed amount of time for each activity.

Once the simulated aircraft has performed a simulated landing and taxied to the parking spot, refueling, cargo upload, and aircraft maintenance are performed concurrently with one exception. Servicing the liquid oxygen (LOX) system is simulated as soon as the aircraft is parked. The simulated servicing lasts a fixed amount of time. No other simulated activities are allowed to occur on the aircraft until LOX servicing is complete. It should be pointed out that other activities could begin as soon as the aircraft is parked as long as they don't occur at the simulated aircraft. The movement of fuel trucks is one example.

Both scheduled and unscheduled maintenance is simulated in the following way. The model assumes that maintenance personnel are always available. For all types of aircraft, scheduled maintenance is always performed and has a fixed duration that is the same for all aircraft and begins as soon as LOX servicing is complete. Unscheduled maintenance duration is simulated as a random length of time whose distribution depends on the type of aircraft simulated. Each aircraft type is assigned a probability of requiring unscheduled maintenance in one of 8 categories: no maintenance, 0-4 hours, 4-8 hours, 8-12 hours, 12-16 hours, 16-24 hours, 25-48 hours, and 48-72 hours. Pseudo-BRACE assumes that repair times are distributed uniformly within each time category. When an aircraft is parked, its unscheduled maintenance time is drawn from the described random distribution. Unscheduled maintenance is performed concurrently with scheduled maintenance.

Two types of aircraft refueling are simulated by Pseudo-BRACE. They are hydrant system refueling and by refueling truck. If an aircraft is parked on parking spot with a fuel hydrant pit, the refueling process is simulated as a hydrant system refueling. If there is more than one refueling pit on the ramp, only one pit can be in operation at any one time. In that case, hydrant refueling is simulated as a first-come first-served (FCFS) queue with one server. A newly arrived aircraft will enter the queue as soon as LOX servicing is complete. Each type of aircraft has a fixed fuel receive rate and a fixed amount of fuel required upon landing. The hydrant system also has a fixed rate that it can pump fuel. The refueling duration time is a fixed time based on the amount of fuel required and the lesser of the two fuel movement rates and a fixed amount of time for hooking up the system.

Aircraft refueling by truck is simulated in the following manner. There are a fixed number of simulated refueling trucks assigned to the airfield. Each truck has the same fixed fuel capacity. They are assigned to aircraft parked on spots without refueling pits on a FCFS basis. Once a truck is assigned to an aircraft it is dedicated to that aircraft until the aircraft has received a full load of fuel. The fuel truck can pump and receive fuel at a fixed rate. Once a truck becomes available, refueling an aircraft with a truck begins by simulating the movement of the truck from the staging area to the aircraft by a fixed amount of time. Once the truck arrives at the aircraft, and LOX servicing is complete, the truck is hooked up to the aircraft, simulated by a fixed amount of time. The refueling duration for a single truck refueling is a fixed time based on the lesser of the amount of fuel required or in the truck with the pumping rate the lesser of the aircraft receive or truck pump rate. When the truck has completed pumping it travels to a single fill stand to refill. The fill stand is simulated as a FCFS queue with a single server. The refill time is simulated as before based on the amount of fuel required to refill the truck to capacity and the lesser of the fuel movement rates. If the aircraft requires more fuel, the truck returns to the aircraft and refuels the aircraft in the manner described above. These activities are repeated until the aircraft has received a full load of fuel.

Only the up-load of cargo is simulated in Pseudo-BRACE. The simulated cargo resources include a finite number of K-loaders, forklifts, and loading docks. An unlimited number of simulated loaded cargo pallets are located in a simulated warehouse. The amount of cargo up-loaded on each aircraft is a fixed number of pallets, each carrying a fixed amount of cargo (in pounds) that depends on the type of aircraft simulated. Upon parking, an aircraft enters a simulated FCFS queue for an available loading dock. Once a loading dock becomes available, any available forklifts move loaded cargo pallets from the warehouse to the dedicated loading dock. The movement time is fixed. A single loading dock is large enough to hold all the pallets for one aircraft. Once the pallets are on the dock, the aircraft enters a simulated FCFS queue for K-loaders. K-loaders can hold up to five pallets each. If more than one K-loader is required to move all the cargo for one aircraft they will be assigned to the aircraft as soon as they become available. The time to load each pallet on a K-loader and the time it takes for a K-loader to get to an aircraft are all simulated by a fixed amount of time. Once a K-loader arrives at an aircraft fixed set-up and separate pallet up-load times are simulated, given that LOX servicing is complete. The first K-loader to arrive at an aircraft incurs an additional fixed "manifest processing" simulated time. This operation continues until the aircraft has received its full cargo load.

Before departing an airfield, each simulated aircraft must spend a minimum amount of time at the airfield, even if all servicing and cargo up-load activities are complete. The minimum time is called a *standard ground time.* This time represents the standard time used by operations planners for scheduling aircrews. Once all servicing has been completed and the standard ground time has been exceeded, aircraft taxi and enter the runway queue. Once the runway becomes available, the aircraft simulates a take-off and departs the system.

Many different performance measures for this simulation are of interest to decision-makers. For the purposes of this research two statistics will be gathered, the mean steady state *turn* time, τ, and the mean steady state *sojourn* time, ρ. Turn time is defined as the amount of time it takes an aircraft, once it is parked, for it to be refueled, have all maintenance completed, and all cargo up-loaded. Sojourn time is defined as the maximum of turn time and standard ground time. To estimate the performance measures n independent replications are performed. These steady-state performance measures are estimated by truncating a constant number of events from the beginning of each simulation replication to eliminate initial transient behavior bias. The reader should interpret all definitions of the following statistics to implicitly include these truncations. Let N represent the number of simulated aircraft that enter the airfield and receive servicing.

To estimate τ let t_{ij} be the amount of simulated turn time for the i^{th} aircraft arriving to the airfield during the j^{th} replication. Then the mean turn time for replication j is

$$T_j = \frac{1}{N}\sum_{i=1}^{N} t_{ij} \quad j = 1, 2, \ldots, n \tag{5.24}$$

so that τ is estimated by

$$\hat{\tau} = \bar{T} = \frac{1}{n}\sum_{j=1}^{n} T_j \tag{5.25}$$

In a similar manner, to estimate ρ, let r_{ij} represent the sojourn time of the i^{th} aircraft to arrive on the airfield during the j^{th} replication. The mean sojourn time for replication j is found by

$$R_j = \frac{1}{N}\sum_{i=1}^{N} r_{ij} \quad j = 1, 2, \ldots, n \tag{5.26}$$

The estimate of ρ is then

$$\hat{\rho} = \bar{R} = \frac{1}{n}\sum_{j=1}^{n} R_j \tag{5.27}$$

5.5.3 Analytical Model. To construct an analytical model, begin by considering a single class capacitated open queuing network depicted in Figure 5.1, with capacity P. This model is based on the Pseudo-BRACE simulation model. In the analytical model, the aircraft are represented as customers that require service at 9 separate stations (queues) in the model. All stations use the FCFS service discipline and the mean service times at each station are derived from the mean times in Pseudo-BRACE.

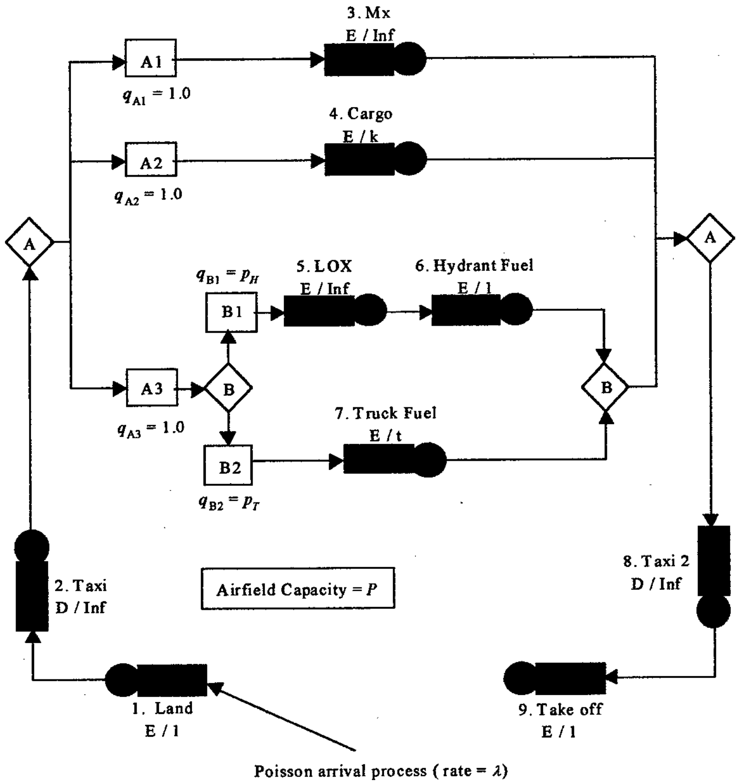

Figure 5.1 Open queueuing network.

The stations model activities that occur within the Pseudo-BRACE model. The mean service time at each station is determined by the mean times required by Pseudo-BRACE to complete the modeled activity. The first station is a single server queue with an exponential service time representing the landing of the aircraft at the airfield. Taxiing to a parking spot is represented by the second station which has an infinite server and a deterministic service time. The third station represents scheduled and unscheduled maintenance with an infinite server and exponential service time. Cargo upload occurs at station 4. The time required to up-load cargo is represented by an exponential random variable and the number of servers is equal to the number of K-loaders in Pseudo-BRACE divided by the number of K-loaders necessary to up-load a single aircraft. Non-concurrent maintenance is modeled by station 5 with an exponential service time and an infinite number of servers. Hydrant refueling is at the sixth station which consists of a single server with an exponential service time. Station 7 models aircraft refueling by truck with the number of servers, t, equal to the number of trucks in the Pseudo-BRACE model. Service time is an exponential random variable. The eighth station models a departing aircraft taxiing to the runway and has an infinite server with a deterministic service time. The ninth and last station models aircraft take-off with a single server queue with an exponential service time.

Due to the concurrent servicing performed in the simulation model, the analytical model contains fork-join constructs depicted as diamonds in Figure 5.1. As described in Chapter II, multiple activities are performed concurrently within the fork-join constructs. The main fork-join node is designated by A with paths $A1, A2$, and $A3$. A second fork-join path is designated as B with paths $B1$ and $B2$. Note that LOX servicing is included in path $B1$. Although in Pseudo-BRACE, LOX servicing must be completed before any other servicing can begin, the movement of the fuel trucks and pallets to the aircraft can begin immediately since it is not possible for them to arrive at the aircraft before LOX servicing is complete. Also, since aircraft maintenance is performed by an infinite server, the time for LOX servicing is added into the total time for maintenance.

In order to apply the MVA algorithm and fork-join heuristic a modification is made to the model. The modified network is shown in Figure 5.2. The open capacitated system is transformed into an equivalent closed network. The transformation is accomplished by adding a new single server queue, station 0, with an exponential service time to the system and setting the number of customers, P, in the system equal to the capacity of the original open network. The new "arrival" station represents that portion of the airlift system that operates outside of the airfield. When all P customers are in the airfield portion of the network, no new arrivals can be generated. This is equivalent to a fully populated open capacitated network. Further, if the airfield is not at capacity, at least one customer is in the arrival queue so that arrivals are generated by a Poisson process with rate of 1 over the mean of the exponential service time. This model does not account for the 2 hour time period that an aircraft will spend waiting for a parking spot in Pseudo-BRACE.

The assumption of the analytical model is that any arriving aircraft that finds the airfield full will divert. Also, hazardous cargo and the separate hazardous cargo parking spots are not modeled.

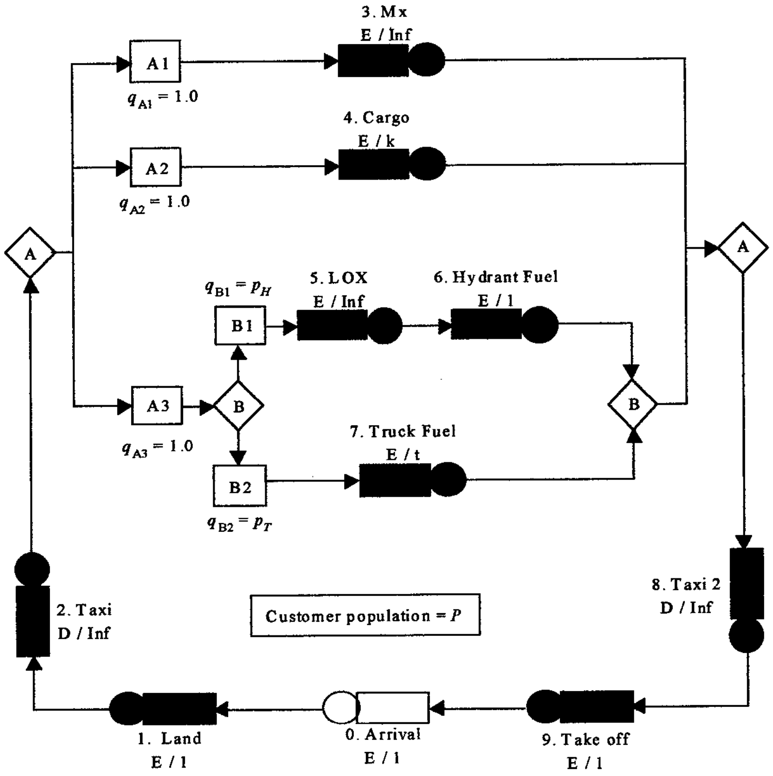

Figure 5.2 Closed queueing network.

Recall from Chapter II that to apply the fork-join heuristic, the MVA algorithm is modified by applying the fork-join approximations and conditioning on the sets of fork-join paths taken [17]. From Figure 5.2 it is seen that the path transit times for fork-join node B when N customers are in the network are given by

$$E\left[T_{B1}(N)\right] = R_5(N) + R_6(N) \tag{5.28}$$

$$E\left[T_{B2}(N)\right] = R_7(N) \tag{5.29}$$

where $R_i(N),\ i = 0, 1, 2, \ldots, 9$ are the station response times found by the MVA algorithm when N customers are in the network. By approximation 2, $T_j(N)$ is approximated by an exponential random variable with rate parameter $\theta_i(N) = 1/E[T_j(N)]$. Using approximation 2, the mean holding time for fork-join node B is given by [17]

$$E[T_B(N)] \approx p_H \left\{ \frac{1}{\theta_{B1}(N)} \right\} + p_T \left\{ \frac{1}{\theta_{B2}(N)} \right\} \tag{5.30}$$

In the same way, path transit times for fork-join node A are approximated by

$$E[T_{A1}(N)] = R_3(N) \tag{5.31}$$

$$E[T_{A2}(N)] = R_4(N) \tag{5.32}$$

$$E[T_{A3}(N)] = E[T_B(N)] \tag{5.33}$$

Mean holding time for fork-join node A is given by

$$\begin{aligned} E[T_A(N)] \approx\ & (1) \left\{ \frac{1}{\theta_{A1}(N)} + \frac{1}{\theta_{A1}(N)} + \frac{1}{\theta_{A3}(N)} \right. \\ & - \frac{1}{\theta_{A1}(N) + \theta_{A2}(N)} - \frac{1}{\theta_{A1}(N) + \theta_{A3}(N)} - \frac{1}{\theta_{A2}(N) + \theta_{A3}(N)} \\ & \left. + \frac{1}{\theta_{A1}(N) + \theta_{A2}(N) + \theta_{A3}(N)} \right\} \end{aligned} \tag{5.34}$$

Applying approximation 1, the cycle time for the network can then be computed by the MVA algorithm for N as [17]

$$CT_0(N) \approx \sum_{i=1}^{2} R_i(N) + T_A(N) + \sum_{i=8}^{9} R_i(N) \tag{5.35}$$

An additional modification is made to the model, depicted in Figure 5.3, by adding station H. Station H represents the standard ground time, SGT, each aircraft must spend on the ground.

This station has an infinite number of servers and a deterministic service time. It is placed on path AH and the solution to the fork-join heuristic must be changed to account for its behavior. If an aircraft traverses all other paths in fork-join node A in less time than SGT, the time spent in fork-join node A is equal to SGT. On the other hand, if the maximum time spent in one, or all, of the other paths is $a > SGT$, the time spent in fork-join node A is a. Using the *memoryless* property of the exponential random variable, and conditioning on all possible outcomes, the fork-join heuristic is easily modified to account for station AH.

Consider the 3 exponential random variables $T_{A1}(N), T_{A2}(N)$, and $T_{A3}(N)$ that represent the time an aircraft clone spends in paths $A1, A2$, and $A3$ respectively. Let $\psi_{Ai}(N)$ be the probability that $T_{Ai}(N) \geq SGT$, $i = 1, 2, 3$, which is found by

$$\psi_{Ai}(N) = 1 - \int_0^{SGT} \theta_{A1}(N) exp\{-\theta_{A1}(N)t\}\, dt = exp\{-\theta_{A1}(N)SGT\} \tag{5.36}$$

It is possible that an aircraft clone spends more than SGT in none, all, one, or several of the three A paths. Let Ω_A be the union of all subsets of possible paths greater than SGT, where there are $2^3 = 8$ different subsets. The subsets are represented by S_j, $j = 1, 2, \ldots, 8$. Let π_j be the probability that subset S_j occurs. Since each path is independent of all the others, π_j is easily calculated by

$$\pi_j = \prod_{i \in S_j} \psi_{Ai} \prod_{i \notin S_j} (1 - \psi_{Aj}) \tag{5.37}$$

Given that a particular S_j occurs, the mean holding time is adjusted in the following manner. Recall that computation of mean holding time is mathematically equivalent to determining the mean time to failure for a parallel system of independent components with exponentially distributed failure times [17]. By the memoryless property of the exponential distribution, the expected value of an exponential random variable given that its value is greater than b is simply b plus the uncon-

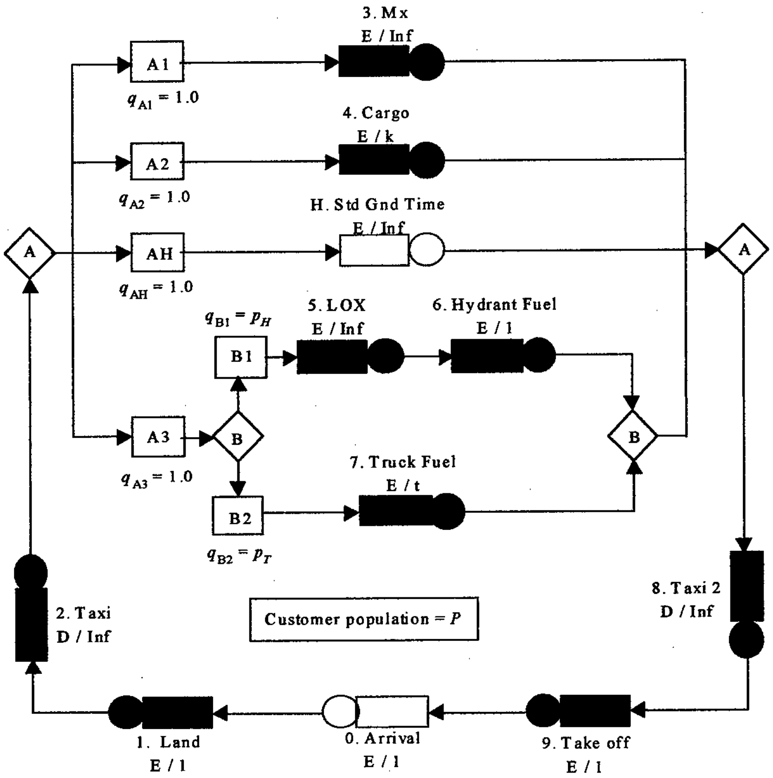

Figure 5.3 Closed queueing network with standard ground time station H

ditional expected value [43]. Therefore, assuming that some of the components have not failed by a given time b, the expected time until failure is b plus unconditional mean time until failure. Hence, for this network, mean holding time for path S_j is designated by $E[\max_{i \in S_j} \{T_{Ai}(N)_{HOLD}\}]$ and is given by

$$E[\max_{i \in S_j} \{T_{Ai}(N)_{HOLD}\}] = SGT + E[\max_{i \in S_j} \{T_{Ai}(N)\}] \tag{5.38}$$

where $E[\max_{i\in S_j}\{T_{Ai}(N)\}]$ is calculated as before. Then the mean holding time for fork-join path is found by conditioning on the different possible paths by

$$\begin{aligned} E[T_A(N)_{HOLD}] &= [\pi_1, \pi_2, \ldots \pi_8]\Big[E[\max_{i\in S_1}\{T_{Ai}(N)_{HOLD}\}], E[\max_{i\in S_2}\{T_{Ai}(N)\}_{HOLD}], \\ &\quad \ldots, E[\max_{i\in S_8}\{T_{Ai}(N)_{HOLD}\}]\Big]^T \end{aligned} \tag{5.39}$$

Then to complete the MVA algorithm, the cycle time for the network is computed as

$$CT_0(N) \approx \sum_{i=1}^{2} R_i(N) + T_A(N)_{HOLD} + \sum_{i=8}^{9} R_i(N) \tag{5.40}$$

The performance measures for the analytical model that correspond to the simulation performance measures turn time and sojourn time are $T_A(P)$ and $T_A(P)_{HOLD}$ respectively, where P is the total number of parking spots in Pseudo-BRACE. To generate the necessary ACV's let $\bar{\mathbf{Y}}^S_{A(j)} = (\bar{Y}^S_{A(1j)}, \bar{Y}^S_{A(2j)}, \bar{Y}^S_{A(3j)}, \bar{Y}_{A(4j)})$, where $\bar{Y}^S_{A(1j)}$ is the sample mean aircraft interarrival time, $\bar{Y}^S_{A(2j)}$ is the sample mean maintenance time, $\bar{Y}^S_{A(3j)}$ is the observed proportion of aircraft that are refueled by hydrants, and $\bar{Y}^S_{A(4j)}$ is the sample mean cargo up-load time for replication j. No other variables need be passed to the analytical model since the other activities simulated in Pseudo-BRACE last a fixed amount of time. Due to the nature of Pseudo-BRACE, the expected values and variances of $\bar{Y}^S_{A(1j)}$ and $\bar{Y}^S_{A(2j)}$ are known since they are functions of input stochastic processes. On the other hand, those of $\bar{Y}^S_{A(3j)}$ and $\bar{Y}^S_{A(2j)}$ are not known since they are strictly outputs of the simulation model. Further, it is known that the interarrival times and maintenance times are independent of each other. All other covariances are unknown. These observed means are then used to calculate $T_A(P)$ (which is not used later in the MVA algorithm due to the nature of the final network) and $T_A(P)_{HOLD}$.

5.6 Performance Comparisons

5.6.1 Experimental Procedures. Experiments are conducted to compare the different methods of estimating the ACV mean. One network design point is selected. The selection of parameters is discussed in a later section. Two types of studies are performed in order to compare the different random vector generation schemes. For each experiment in both studies, 1,000 random vectors are generated to find $\hat{\mu}_Z$ by each generation scheme. The first study consists of 50 experiments that compare the different approximated values of $\hat{\mu}_Z$ with the actual value of μ_Z. For each experiment, 20 consecutive replications are chosen at random from a set of 10,000 replications. The 20 replications are used to estimate parameters or for re-sampling for each of the different random vector generation schemes. Comparisons are made using relative absolute error percentage, coverage, and MSE. The second study compares the coverage and MSE observed for controlled estimates of the performance measures using 25 experiments with the number of simulation model replications equal to 20. In this case, the observed data from the appropriate set of simulation replications is used to generate the necessary random vectors.

Comparisons for the first study are made in the following manner. Let μ_Z be the actual expected value of the ACV. For the 1,000 random vectors generated for the h^{th} experiment, $h = 1, 2, \ldots, 50$, an estimate of μ_Z is computed. Call the estimate $\hat{\mu}_{Z(h)}(l)$, where $l = 1, 2, \ldots, 6$, denote the 6 different random vector generation schemes used. The methods are described later in the next section. Then relative absolute error, $E_h(l)$ for the h^{th} experiment using the l^{th} scheme is calculated by

$$E_h(l) = \left| \frac{\hat{\mu}_{Z(h)}(l) - \mu_Z}{\mu_Z} \right| \tag{5.41}$$

The relative absolute error percentage for method l is then given by

$$E(l)\% = (100\%)\frac{1}{50}\sum_{h=1}^{50} E_h(l) \quad l = 1, 2, \ldots, 6 \tag{5.42}$$

Another means of assessing the accuracy of the different schemes for approximating the ACV mean is to compare realized coverage for each method. Begin by letting $\hat{\sigma}_h^2(l)$ denote the 6 different estimates of the variance of $\hat{\mu}_{Z(h)}(l)$. For the h^{th} experiment, the confidence interval estimate is given by

$$\hat{\Lambda}_h(l) = \hat{\mu}_{Z(h)}(l) \pm \hat{H}_h(l) \tag{5.43}$$

where $\hat{H}_h(l)$ is the estimated half-width given by

$$\hat{H}_h(l) = t_{1-\alpha/2,999}\sqrt{\frac{\hat{\sigma}_h^2(l)}{1000}} \tag{5.44}$$

with $\alpha = 0.10$. The estimated confidence interval coverage probability is found by first letting

$$\hat{I}_h(l) = \begin{cases} 1 & \text{if } \mu_Z \in \hat{H}_h(l) \\ 0 & \text{otherwise} \end{cases} \tag{5.45}$$

for $l = 1, 2, \ldots, 6$ and $h = 1, 2, \ldots, 50$. The estimate of the confidence interval coverage is given by the calculated coverage fraction for $\hat{\Lambda}_h(l)$, computed as

$$\hat{I}(l) = \frac{1}{50}\sum_{h=1}^{50} \hat{I}_h(l) \quad l = 1, 2, \ldots, 6 \tag{5.46}$$

Realized coverage is not always completely indicative of the accuracy of an approximation. For example a point estimate may be very close to μ_Z, but if the associated confidence interval is small, coverage may not be realized. Another measure of accuracy that considers this is the estimated value of the MSE of point estimator. The estimated MSE is computed by

$$\widehat{MSE}(l) = \frac{1}{h}\sum_{h=1}^{50}\left(\hat{\mu}_{Z(h)}(l) - \mu_Z\right)^2 \quad l = 1, 2, \ldots, 6 \tag{5.47}$$

The actual expected values of the ACV for both turn time and sojourn time are estimated by performing 10,000 replications of Pseudo-BRACE and finding the ACV for each of those replications. The resulting sample means are used to estimate μ_Z for both performance measures.

The second study compares estimated coverage and MSE for the different schemes, but in this case the comparison is made using the ACV controlled performance measures. For each of the 25 experiments, the 6 different estimates of μ_Z generated by the different methods are used to produce the controlled estimates. In addition, to achieve the 10% confidence interval size for $\hat{\mu}_Z$ compared to the controlled response, 4,000 random vectors are generated for each experiment. The coverage and MSE comparisons are then made using the same procedures outlined above with $d = 25$ and using μ, the expected value of the performance measure, in place of μ_Z. Also, $\hat{\mu}_h(l)$, the controlled estimate of μ for the h^{th} experiment using the l^{th} different random vector generation scheme, is used instead of $\hat{\mu}_{Z(h)}(l)$. Further the confidence interval half width is estimated using Equation (2.21). The expected values of the performance measures are estimated using the same 10,000 replications described above.

5.6.2 Random Vector Generation Schemes. The following random vector generation schemes are used. Each is based on the methods described in Sections 5.2, 5.3, and 5.4. Recall that the input to the analytical model, that must be approximated by these schemes, is $\bar{\mathbf{Y}}^S_{A(j)}$ for replication j where $\bar{Y}^S_{A(1j)}$ is the sample mean aircraft interarrival time, $\bar{Y}^S_{A(2j)}$ is the sample mean maintenance time, $\bar{Y}^S_{A(3j)}$ is the observed proportion of aircraft that are refueled by hydrants, and $\bar{Y}^S_{A(4j)}$ is the sample mean cargo up-load time. The first scheme examined is the non-parametric SIMDAT method. The parametric method described in Section 5.3 is used to create 3 different schemes for generating random vectors that depend on our knowledge of the random variate parameters. The first is called the NORM-EST method since we assume all parameters of the multivariate normal distribution are unknown and are thus estimated by the observed data. On the other hand, NORM-MU assumes that we know means of $\bar{Y}^S_{A(1j)}$ and $\bar{Y}^S_{A(2j)}$. All other means and all values of the covariance matrix are then estimated. NORM-ALL uses all known parameters—the means and variances of aircraft interarrival and maintenance times and the zero covariance values between these two variables—when generating random vectors. All other parameters are estimated.

The combination methods used are described in Section 5.4. They are referred to here as the NORM-BOOT and the NORM-SIMDAT. In both combined methods, $\bar{Y}^S_{A(1j)}$ and $\bar{Y}^S_{A(2j)}$ are generated via the parametric method using the known parameters, and $\bar{Y}^S_{A(3j)}$ and $\bar{Y}^S_{A(4j)}$ are generated with a non-parametric method. Also, both combined methods assume that $\bar{Y}^S_{A(1j)}$ and $\bar{Y}^S_{A(2j)}$ are independent of $\bar{Y}^S_{A(3j)}$ and $\bar{Y}^S_{A(4j)}$ even though this is not true. For the NORM-BOOT, it is further assumed that $\bar{Y}^S_{A(3j)}$ and $\bar{Y}^S_{A(4j)}$ are independent of each other. Based on that assumption, the bootstrap method is used to re-sample the observed data and generate new random vectors. The NORM-SIMDAT uses the SIMDAT method to generate new pseudo-data points for $\bar{Y}^S_{A(3j)}$ and $\bar{Y}^S_{A(4j)}$. In that case, no assumption about their dependence structure is necessary.

5.6.3 Network Settings. One experimental design setting is used to conduct the experiments described above. Table 5.1 describes the number and types of resources used in Pseudo-BRACE. Other network settings include the length of the standard ground time, 2.25 hours, and the length of time an aircraft will wait before diverting, 2 hours. Recall that an aircraft without hazardous cargo will be parked at a spot with a fuel hydrant pit, if one is available. If not, it will be parked first at a non-hazardous parking spot and then a hazardous parking spot. The probability that an arriving aircraft will up-load hazardous cargo is 0.10.

Aircraft arrive to the airfield via a simulated Poisson process with mean arrival rate of 1.25 aircraft every hour. One thousand aircraft arrive at the airfield during each simulation replication.

Table 5.1 Pseudo-BRACE resources.

Resource	Number
Total Parking spots	12
Hydrant parking spots	4
Hazardous parking spots	4
Fuel Trucks	8
Fuel fill stand	1
K-Loaders	8
Forklifts	19
Loading docks	10

To eliminate the effects of initial transient bias, statistics are gathered on the last 900 arrivals. The different aircraft types and their parameters are listed below in Table 5.2. Included in Table 5.2 are the probabilities that an aircraft arrival will be of a particular type. Based on the fuel load requirements and fuel truck capacities, fuel trucks will have to make 2 trips to an aircraft to refuel it. The cargo up-load requirement and average pallet cargo weight results in a total of 12 pallets for every aircraft up-load. This translates to 3 K-loaders (K-loader capacity equals 5 pallets) for every aircraft. The aircraft unscheduled maintenance category probabilities are recorded in Table 5.3. Within each category, the unscheduled maintenance time is uniformly distributed. Scheduled maintenance (performed concurrently with unscheduled maintenance) is fixed at 30 minutes and LOX servicing is fixed at 9 minutes.

Table 5.2 Pseudo-BRACE aircraft parameters.

Type	Probability	Fuel load	Cargo load	Fuel receive
		(gals)	(tons)	rate (gpm)
C-130	0.30	10000	26.0	450.0
C-17	0.20	10000	26.0	450.0
C-5	0.20	10000	26.0	450.0
C-141	0.15	10000	26.0	450.0
B-747	0.05	10000	26.0	450.0
DC-8	0.10	10000	26.0	450.0

The known mean and variance for maintenance times are given by $E[\bar{Y}^S_{A(1j)}] = 1.6921$ hours, $Var[\bar{Y}^S_{A(1j)}] = 0.03327$ hours^2. The maintenance time includes LOX servicing and scheduled and unscheduled maintenance. The mean and variance for the sample mean interarrival times are

Table 5.3 Pseudo-BRACE unscheduled maintenance probabilities.

	Time category (hours)							
Type	0-4	4-8	8-12	12-16	16-24	24-48	48-72	None
C-130	0.000	0.000	0.000	0.000	0.000	0.000	0.000	1.000
C-17	0.000	0.010	0.000	0.000	0.010	0.000	0.010	0.930
C-5	0.043	0.057	0.029	0.036	0.021	0.021	0.007	0.786
C-141	0.033	0.030	0.027	0.033	0.017	0.020	0.003	0.837
B-747	0.000	0.000	0.000	0.000	0.000	0.000	0.000	1.000
DC-8	0.000	0.000	0.000	0.000	0.000	0.000	0.000	1.000

$E[\bar{Y}^S_{A(2j)}] = 0.8$ hours and $Var[\bar{Y}^S_{A(2j)}] = 0.0007111$ hours^2. Also, it is known that $\bar{Y}^S_{A(1j)}$ and $\bar{Y}^S_{A(2j)}$ are independent of each other. The other elements of the covariance matrix are unknown.

5.6.4 Results. The first study results indicate that the different schemes have varying levels of success in approximating μ_Z. The results are reported in tables 5.4 and 5.5. The relative absolute percentage error and MSE indicate approximations that are very close to the actual ACV mean for all methods explored. However, the coverage estimates are well below the nominal value of 0.90. This is an indication that the confidence intervals are very small when compared to the size of μ_Z. In fact, the reported average confidence interval width for all methods for both performance measures are all approximately 0.01 while the values of μ_Z are about 2.9 and 3.3 for turn time and sojourn time respectively. The most striking outcome is the two groups formed based on results. One group consists of the methods that estimate all parameters and the other group is formed of the methods that use some known parameters. The results within each group are very similar. The group where all parameters are estimated, which are SIMDAT and NORM-EST, is also the group that produces the largest errors and smallest coverage estimates. The other group that uses known parameters performs markedly better.

Results from the second study appear in Tables 5.6 and 5.7. The Tables indicate that the grouping of the methods based on results is the same as that of the first study. The methods that use known parameters perform demonstrably better with the two combined methods providing marginally better results than the completely parametric methods. Note that the coverage estimates

Table 5.4 ACV mean approximation, $\hat{\mu}_Z$, comparisons (Turn time).

Turn time				
Method	Mean	Relative absolute error (%)	Coverage (%)	MSE ($\times 10^{-4}$)
SIMDAT	2.8671	2.79	0.14	11.6518
NORM-EST	2.8721	2.69	0.18	12.1099
NORM-MU	2.8731	0.79	0.52	1.0418
NORM-ALL	2.8719	0.78	0.50	0.7895
NORM-BOOT	2.8709	0.67	0.52	0.6916
NORM-SIMDAT	2.8694	0.72	0.54	0.7717
Actual Mean	2.8712			

Table 5.5 ACV mean approximation, $\hat{\mu}_Z$, comparisons (Sojourn Time).

Sojourn time				
Method	Mean	Relative absolute error (%)	Coverage (%)	MSE ($\times 10^{-4}$)
SIMDAT	3.2445	2.22	0.12	7.4457
NORM-EST	3.2493	2.15	0.20	7.7651
NORM-MU	3.2499	0.64	0.54	0.6843
NORM-ALL	3.2490	0.61	0.52	0.4778
NORM-BOOT	3.2479	0.53	0.56	0.4281
NORM-SIMDAT	3.2467	0.56	0.52	0.4801
Actual Mean	3.2484			

are slightly lower than nominal for some of the better performing methods. This is attributable to the extremely small realized confidence interval widths for the controlled responses of approximately 0.044 for turn time and 0.052 for sojourn time. Further, the realized coverage for the uncontrolled responses is only 84% for both responses. Hence, in some cases, the ACV method actually increases the realized coverage. Therefore, the extremely small MSE is a better indicator of the small observable bias in the better performing methods. To achieve the same results for SIMDAT and NORM-EST additional random vectors need to be generated. Also, it should be noted that the ACV performs well in reducing variance and the resulting confidence interval widths. For turn time, the confidence interval width reduction is approximately 70% and 61% for sojourn time.

Table 5.6 Controlled response comparisons (Turn time).

Turn time			
Method	Mean	Coverage (%)	MSE ($\times 10^{-4}$)
SIMDAT	2.6825	0.40	35.5351
NORM-EST	2.6857	0.40	32.8867
NORM-MU	2.6613	0.70	3.6922
NORM-ALL	2.6624	0.85	3.2714
NORM-BOOT	2.6582	0.75	3.3086
NORM-SIMDAT	2.6597	0.80	2.9380
Actual Mean	2.6556		

Table 5.7 Controlled response comparisons (Sojourn time).

Sojourn time			
Method	Mean	Coverage (%)	MSE ($\times 10^{-4}$)
SIMDAT	3.2798	0.50	30.0825
NORM-EST	3.2838	0.50	28.4189
NORM-MU	3.2620	0.95	3.1823
NORM-ALL	3.2632	0.95	3.9197
NORM-BOOT	3.2589	0.90	2.4078
NORM-SIMDAT	3.2601	0.95	2.0298
Actual Mean	3.2563		

5.7 Conclusion

This research into different non-parametric and parametric methods for approximating the mean of the ACV confirms that the methods examined have merit and can be used to reduce the observed bias in ACV controlled responses. Despite the lack of knowledge of certain parameters of $\bar{\mathbf{Y}}^{S}_{A(j)}$, this research has demonstrated several methods that generate random vectors that mimic $\bar{\mathbf{Y}}^{S}_{A(j)}$ for the Psuedo-BRACE simulation model. These methods allow us to continue research into other areas that rely upon the ACV method—namely surrogate search—using the Psuedo-BRACE simulation model.

VI. Surrogate Search Methods

6.1 Overview

We have shown that the ACV Monte Carlo method efficiently reduces the variance and confidence interval width of simulation performance measures for two different simulation models. Hence the potential of reducing simulation study times using an external analytical model is demonstrated with the application of the methods discussed thus far. The next step is to show how the ACV method of variance reduction can be used as a starting point for a new method of reducing simulation study times. The new method consists of computing performance measures from an analytical model instead of performing simulation replications in order to estimate system performance measures. Since successful ACV application requires an analytical model that can be computed much more rapidly than the simulation model, significant time savings can be achieved.

We demonstrate a method for validating and using an analytical model as a surrogate of a simulation model to perform *surrogate searches* of a simulation experimental design space. We use response surface methodology (RSM) as the context for the surrogate search method. It is important to point out, that the intention is not to eliminate the simulation model. Rather, the goal is to augment, or enhance, the performance of the simulation model with an analytical model that has been validated for the specific purpose of performing the surrogate search. In general, a valid model is one that is an accurate representation of the system under study [32]. The specific definition of analytical model validation for the purposes of performing a surrogate search is described below in section 6.3. The validation process is performed in conjunction with the application of the ACV method during as a RSM simulation study design of experiment is realized.

During a surrogate search, the analytical model performs a search of a design space where the expected responses of the simulation model are unknown, in order to identify *interesting* (locally optimal and/or nearly optimal) design points for further investigation. Using the surrogate search results as a guide, the simulation model is replicated at these interesting points in order to validate the surrogate search results. If the analytical model is an accurate predictor of the simulation model, only those validation replications are necessary in order to establish a new experimental design for the simulation study. Significant time savings are then realized as it is not necessary to perform costly simulation replications at *uninteresting* points along the search path.

The chapter is organized in the following manner. The first section briefly describes a common simulation model verification and validation process. This process is modified in the following section as a new validation process in order to justify (validate) an analytical model for the surrogate search method. The final section of the chapter develops the surrogate search method.

6.2 Simulation Model Verification and Validation

Simulation models are used throughout industry and the military to provide decision-makers with the information necessary to make informed decisions concerning complex systems and problems. The correctness of these decisions depends on the accuracy of the simulation models. Verification and validation methods are employed by simulation developers and analysts in order to determine the accuracy of simulation models [45]. Several authors have proposed definitions for verification and validation. Two of the more commonly accepted definitions for model validation follow [32, 45]. Model *validation* is often defined as "substantiation that a computerized model within its domain of applicability possesses a satisfactory range of accuracy consistent with the intended application of the model" [47]. Similarly, "validation is concerned with determining whether the conceptual simulation model (as opposed to the computer program) is an accurate representation of the system under study. If a model is valid, then the decisions made with the model should be similar to those that would be made by physically experimenting with the system (if this were possible)" [22]. On the other hand, *verification* ensures that the valid conceptual model is correctly translated into a computer program that performs as intended [32]. A related term often encountered in verification and validation discussions is model *credibility.* A model is considered credible if the using decision-maker accepts the model as valid and uses it to make and implement decisions [15].

There are many important issues to consider when performing simulation model verification and validation. We describe several of these issues that should be considered when we develop the surrogate search validation process in the following section. The first consideration is that a simulation model should be developed and validated for a specific purpose [32, 45]. The purpose could be to answer a single or several questions and each of these questions must be addressed separately. Analysts should also realize that a simulation model is only an approximation of an actual system and can never be made absolutely valid [32]. A model can be made more valid by spending more time and money on the validation process, however the analyst should consider the cost-effectiveness of the additional expenditures required to increase model validity. Again, the issue of model applicability is the primary consideration when attempting to assess when a model is "valid enough" [45]. Finally, model verification and validation should be an integral part of the model development process [15, 32, 45].

The rest of this section describes the simplified verification and validation process presented by Sargent [45]. The topics addressed include descriptions of an integrated validation process, data validity, conceptual model validity, computerized model validity, and finally operational validity.

6.2.1 Validation Process. Sargent states that there are three basic approaches to determining simulation model validity [45]. All three approaches require that the model development team perform verification and validation during the model development process. The model development team makes the model validity decision in the first approach using a subjective method based on the results of numerous tests and evaluations performed during model development. To conduct the next approach, an independent verification and validation (IV&V), a third party independent of the model development team and the end-user of the model decides if the model is valid. This is accomplished after the model has been developed. The independent party makes a subjective decision based on their own evaluation of the delivered model. A scoring model is used for the final verification and validation technique (see Balci [4]). In this method, scores are determined subjectively for various aspects of the validation process, which are combined in an overall score for the simulation model. The model is then considered valid if the score exceeds some pre-determined score. Sargent [45] notes that the scoring method is rarely used in practice and discourages its use for determining model validity. He states that the subjectiveness of this approach is often masked, making the method appear to be objective. He goes on to describe several other technical deficiencies of the method as well.

Both Sargent [45] and Law and Kelton [32], as well as other researchers, make it clear that model verification and validation should be an integral part of model development. To understand that relationship, we now describe the simplified modeling process presented by Sargent [45]. A diagram of the modeling process appears in Figure 6.1. The *problem entity* is the system (idea, situation, etc.) that is to be modeled. The problem entity can either currently exist or be a new proposed entity that doesn't actually exist in the real world. The mathematical/logical/verbal representation of the problem entity is called the *conceptual model*, and the *computerized model* is the computerized implementation of the conceptual model. The dashed lines that connect each item are the phases of the modeling development process. The conceptual model is developed during the *analysis and modeling phase*, the conceptual model is translated into a computerized model during the *computer programming and implementation phase*, and finally experiments are performed using the computerized model during the *experimentation phase* in order to make inferences about the problem entity.

Figure 6.2 illustrates how the verification and validation process is integrated into the model development process [45]. *Conceptual model validity* is established by examining the theories and assumptions that are used to develop the conceptual model. This results in a conceptual model that is a valid representation of the problem entity for the particular set of questions being asked. Once

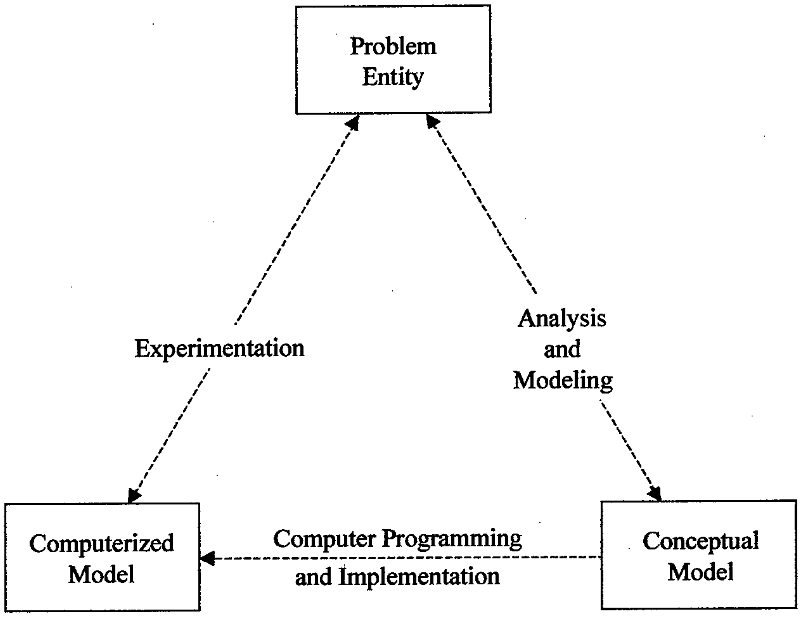

Figure 6.1 Simplified simulation model development process. Adapted from Sargent [45].

this is established, *computerized model verification* is performed to ensure that the computer code correctly implements the conceptual model. *Operational validity* is establishing the computerized model output accurately represents the problem entity for the region of intended purpose. Finally *data validity* is defined as determining that the data required to accomplish each of these tasks is correct and adequate. Sargent [45] points out that model development is an iterative process and that each of the verification and validation steps must be performed for each iteration.

Numerous validation techniques are available to the analyst, however no algorithm or theorem exists for determining the best technique for any given situation [45]. Some of these techniques are described below with some of the attributes that effect their utility. See for example Sargent [45] and Balci [5] for a more complete list.

Comparison to Other Models The simulation model output is compared to the output from another *valid* model. These other valid models could be analytical models or some other valid and credible simulation model.

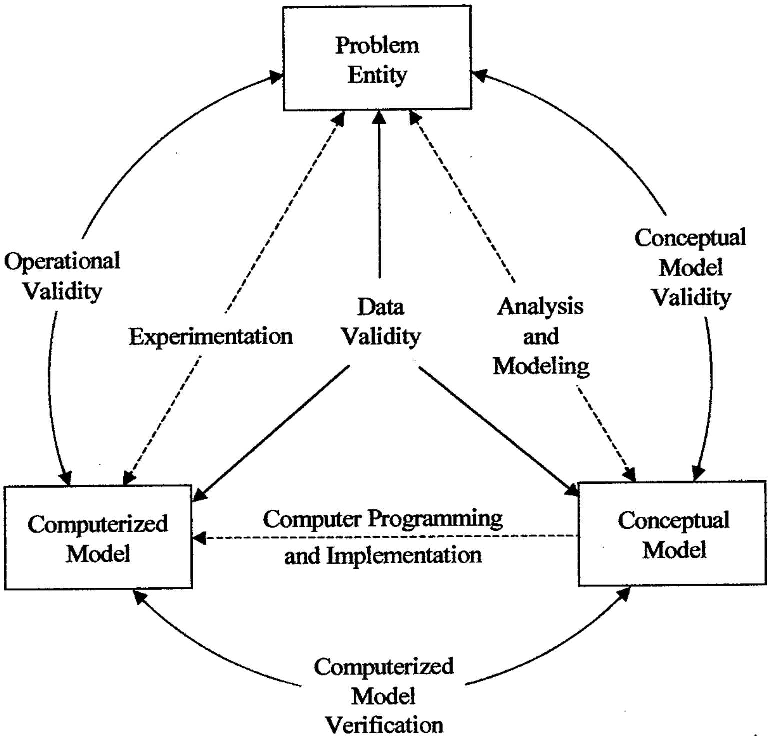

Figure 6.2 Integrated simulation model verification and validation process. From Sargent [45].

Face Validity This technique involves asking system experts if the model and/or its behavior is reasonable.

Historical Data Validation Given that historical data exists, part of the data can be used to develop the model and the remainder of the data can be used to test the validity of the model.

Turing Test Experts examine output from the model and problem entity and try to determine the source. The better the model, the more difficult it is for the experts to differentiate the output.

Predictive Validation System behavior predictions made by the model are compared to actual system behavior. The system data could come from an operational system or experiments made on the system.

The remainder of this section examines the verification and validation steps in more detail.

6.2.2 Data Validity. Valid data is required to construct the conceptual model, validate the computerized model and to perform experiments on the validated model. Model validation is concerned with the first two categories of data. Sargent [45] points out that data validity in normally not considered part of the model validation process. However, valid data is vital to the construction of a valid model. The problem is that obtaining enough valid data is a difficult, time consuming task and the lack of it is a major contributor to failed model building attempts.

6.2.3 Conceptual Model Validation. Conceptual model validation is achieved by ensuring that the theories and assumptions used to create the model are correct and that the representation of the problem entity, including the model's structure, logic and mathematical relationships, are "reasonable" for the intended purpose of the model. Sargent [45] suggests that face validation is one of the primary validation techniques used for conceptual model validation. Using flowcharts, graphical models, and the set of mathematical equations used in the model, the analysts and system experts can determine if the conceptual model possess face validity for its intended purpose.

6.2.4 Computerized Model Verification. Computerized model verification ensures that the computerized simulation model is performing as expected. This can be an arduous and difficult task, particularly for complex models. Many techniques are available to the development team to ensure that the conceptual model has been correctly translated into computer code and that the code is free of bugs. However, since the primary purpose of this exposition is to establish a frame of reference for establishing a surrogate search validation process, these methods will not be discussed here. Suffice to say, that computerized model verification is a continuous process that should occur throughout the model development process [45].

6.2.5 Operational Validity. The final step in the verification and validation process is operational validity. This process examines the computerized model's output to determine if it is accurate enough for its intended purpose. Although this is the last step before declaring a model valid, if operational validity is not achieved the analyst must return to one of the previous steps to determine the cause. Inaccurate output is caused by any combination (singly or together) of an invalid conceptual model, incorrectly programmed computerized model, or invalid data [45].

Any of the validation techniques mentioned in Sargent [45] and Balci [5] are applicable to operational validity. The analyst, system experts, and end-user determine the choice of validation technique and whether the evaluations are subjective or objective. The key to operational validity is whether or not output data can be collected from the problem entity itself. If the output data is available, the computerized model's behavior should be compared to the problem entity. If the system is non-observable, Sargent [45] suggests comparing the model's output to other validated models.

At least two different sets of experimental inputs are required to ensure a high degree of model validity according to Sargent [45]. The outputs at each of the different input sets are compared using either graphs, confidence intervals, or hypothesis tests. Graphs are the most commonly used approach followed by confidence intervals [45].

Graphs can be used to examine the accuracy of a simulation model's accuracy over a range of different input settings. Types of applicable graphs include histograms, box plots, and behavior graphs. Behavior graphs are simply graphs of output data from the simulation model and problem entity over a range of inputs. It is vital that the outputs graphed relate to the intended purpose of the simulation model. The graphs can be subjectively examined by the development team, system experts, or as a Turing test to determine operational validity.

Confidence intervals, simultaneous confidence intervals, and joint confidence regions are also useful validation tools. They can be formed for the difference between the parameters of the model and system or for the distribution of the output values of each over the range of model applicability. Validation decisions can then be made based on the model accuracy observed.

Hypothesis tests can also be used to confirm operational validity. Given an appropriate set of data, the hypothesis to be tested can be stated as [45]:

$$\begin{array}{ll} \mathbf{H}_0: & \text{Model is valid for the acceptable range of accuracy under the set of} \\ & \text{experimental conditions} \\ \mathbf{H}_1: & \text{Model is invalid for the acceptable range of accuracy under the set of} \\ & \text{experimental conditions} \end{array} \qquad (6.1)$$

Detailed procedures for the hypothesis test method of validating simulation models can be found in Balci and Sargent [6] and Banks, Carson, and Nelson [7].

6.2.6 Verification and Validation Summary. Simulation model verification and validation is a difficult, yet vital task that should be integral to the model development process. Further, since simulation models are often used more than once in order to investigate problems other than the initial problem entity, model validation should be performed throughout the model life cycle [5,45]. The verification and validation task is made difficult by the lack of specific algorithms or theorems that specify the best validation method. Indeed, validation can often be a subjective judgement made with little or no data from a real-world problem entity. However, by following the guidelines sketched above, the validation decision will provide the user and analyst with the guidance necessary for appropriate model use.

6.3 Surrogate Search Validation

Significant variance reduction achieved using the ACV method indicates that the analytical model output is correlated to the simulation model output when the inputs to both models are similar. While this suggests that the analytical model might be used as a surrogate to the simulation model, this is not necessarily the case. For example, the two model outputs could be negatively correlated. Or, even though variance reduction for the ACV method is a result of linear correlation, the actual correlation between the two models may more accurately be of a quadratic or higher order correlation. Even if the correlation is positive and linear, variance reduction occurs in small neighborhoods about each of the design points replicated. This tells us little about the behavior of the two models over the entire design region. Despite these shortcomings, the ACV method can be used as an integral part of a formal process for justifying (validating) the use of an analytical model as a surrogate of the simulation model. In fact, the validation process can be integrated into a RSM simulation study using the ACV method in a manner that is nearly transparent.

We develop a validation process of an analytical model for the purposes of conducting a surrogate search in the context of an RSM simulation study that uses the ACV method as a primary validation tool. The process is adapted from the simulation model verification and validation process described above in Section 6.2. The primary difference is that in the above procedure, the simulation model is validated against some "real-world" system, while in this case the analytical model is validated against a previously validated simulation model, not the system under study. When constructing an analytical model in order to generate an ACV this is exactly the frame of reference. While it may be true that the analytical model is (or can be made) valid for the system under study, we are really only concerned with comparing the output of the analytical model with the output of the simulation model, not the real system.

The surrogate search modeling process is illustrated in Figure 6.3. The figure is essentially the same as Figure 6.2. The main differences are that a "Valid/Credible Simulation Model & Problem Entity" replaces the "Problem Entity" in the surrogate search modeling figure and "Operational Validity" is replaced by "Surrogate Search Operational Validity". Further, we explicitly state that the model we are developing and validating is an analytical model. These changes reflect the fact that we are developing an analytical model to serve as a surrogate of the simulation model, not the system under study. Otherwise, the figure, and the process, is identical to that for the simulation modeling process.

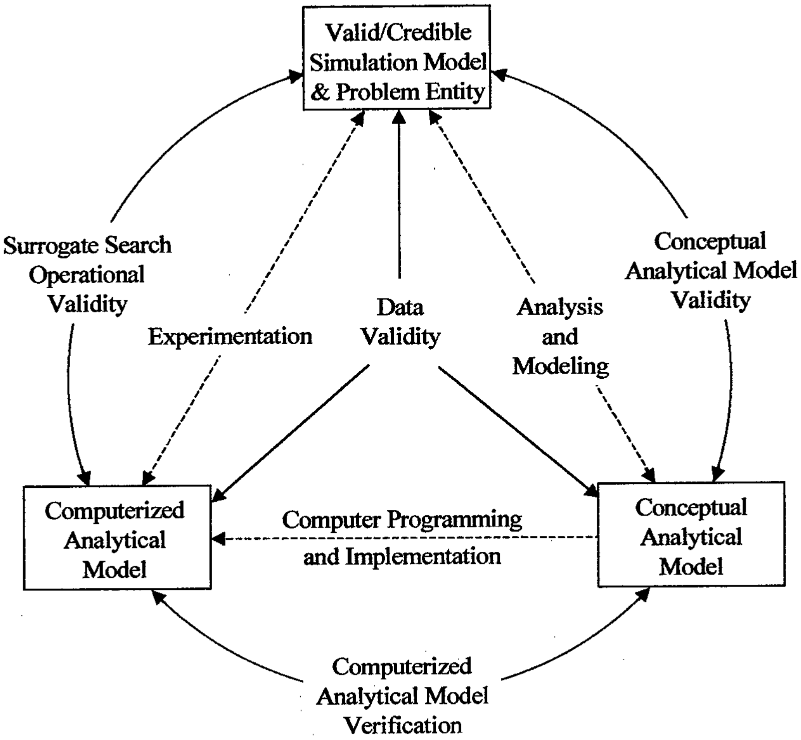
Valid/Credible Simulation Model & Problem Entity
Surrogate Search Operational Validity
Experimentation
Data Validity
Analysis and Modeling
Conceptual Analytical Model Validity
Computerized Analytical Model
Computer Programming and Implementation
Conceptual Analytical Model
Computerized Analytical Model Verification

Figure 6.3 Surrogate search verification and validation process. Adapted from Sargent [45].

Recalling that our overall goal is to reduce simulation study times, we propose a surrogate search validation method that is as streamlined as possible, while still adhering to the validation principles outlined above. This is primarily accomplished by analyzing the results of the ACV method and the response surfaces generated during the initial stages of a RSM simulation study. The validation process is outlined below. We begin with a discussion of *conceptual analytical model validity.* Here we consider the particular modeling requirements necessary for an analytical model to serve as a surrogate for a particular simulation study. We then briefly discuss *computerized analytical verification.* Next we describe a means of achieving *surrogate search operational validity* based on the results of the ACV method and response surface comparisons. This corresponds to achieving operational validity in the simulation validation method above. We do not address data validity for the surrogate search method other than to say it consists of ensuring that the valid data used and produced by the simulation model are correctly collected and interpreted for use in validating the analytical model. This is relatively easy when compared to ensuring data validity for the simulation model, however it is a vital task that must not be ignored. We complete the discussion with a brief summary.

6.3.1 Conceptual Analytical Model Validity. Recall that conceptual model validity is achieved by ensuring that the theories and assumptions used to create the model are correct and that the representation of the problem entity, including the model's structure, logic and mathematical relationships, are "reasonable" for the intended purpose of the model [45]. We achieve conceptual analytical model validity in the same manner, keeping in mind our specific intended purpose of serving as a surrogate of the simulation model for a specific study.

We first consider the theoretical structure and assumptions of the analytical model. Since we are developing a model of an existing valid/credible simulation model, this task can be somewhat simpler than attempting to create a model of a real world system. We already know (or can readily determine) the inputs, parameters, and logic that are used in the simulation model. For example, in Chapter V it is apparent that the conceptual model of the Psuedo-BRACE simulation model is an open network of queues with specific service times and disciplines. The obvious difficulty in translating Psuedo-BRACE into an analytical model is that the defined open queuing network cannot be analytically solved. This is true with most simulation models. If they can be solved analytically there is little, if any, reason to construct the simulation model in the first place. The task then is to develop an analytical model that adequately approximates the behavior of the simulation model. This mirrors the task of developing a conceptual simulation model of a real system, since a simulation model can at best be an approximation of the actual system [32].

As with simulation conceptual model validity, the technique recommended for evaluating the theories and assumptions used to develop the conceptual analytical model is that of face validity [45]. In this case, the system expert will usually be the simulation analyst that regularly uses the simulation model and/or is tasked to perform the specific simulation study. For analytical model reasonableness, some assumptions must be made in order to maintain a model that can be solved analytically. For example, in a simulation model a particular service time may be modeled as being distributed with a Weibull distribution. To maintain tractability, the analytical model may have to model that same service time with an exponential distribution. These differences must be considered when determining the face validity of the analytical conceptual model. The analyst must remember that we are attempting to approximate the simulation model, not solve it. Additionally, the computational effort required to compute the analytical model should be considered as well. The whole point of this exercise is to develop a surrogate of the simulation model that can be rapidly evaluated in order to save time.

We now turn to the specific intended purpose of the analytical model—to serve as a surrogate of the simulation model for a specific simulation study. To perform as a surrogate it must meet two requirements:

1. The conceptual analytical model must have an output(s) that corresponds to the output(s) of interest for the simulation study.
2. The conceptual analytical model must be able to adjust the factors that are adjusted by the simulation model during a simulation study.

Hence there are specific mapping requirements put on some of the outputs and inputs of the conceptual analytical model, based on the specific simulation study to be performed. The following discussion describes in detail the mappings between the two models that must occur.

Recall from Chapter III that a simulation model can be represented as the function f^S for replication i as

$$f^S\left(\boldsymbol{\phi}^S, \mathbf{V}_i\right) = f^S\left(\boldsymbol{\phi}^S, g(\boldsymbol{\theta}^S)_i\right) = \mathbf{Y}_i^S \tag{6.2}$$

where $\boldsymbol{\phi}^S$ is the vector of structural parameters, $\boldsymbol{\theta}^S$ is the vector of random variate parameters, $g(\cdot)$ is the random variate generator, and $g(\boldsymbol{\theta}^S)_i = \mathbf{V}_i$ is the vector of random variate processes that drive the simulation model for replication i. The output vector $\mathbf{Y}_i^S$ consists of a vector of stochastic processes realized during the ith replication. The processes include the performance measure of interest for the particular simulation study as well as the realized "input" processes. As before, let $\mathbf{Y}_P^S$ represent the output stochastic process of interest with $\bar{Y}_P^S$ the appropriate statistical estimate of the performance measure formed upon completion of the planned replications.

In a similar manner, the analytical model can be represented as the function

$$f^A = f^A\left(\boldsymbol{\phi}^A, \boldsymbol{\theta}^A\right) = \mathbf{Y}^A \tag{6.3}$$

where $\boldsymbol{\phi}^A$ is the structural parameter vector and $\boldsymbol{\theta}^A$ is the random variable parameter vector of the analytical model. These vectors are defined in a manner similar to that for the simulation model. The structural parameter vector, $\boldsymbol{\phi}^A$, consists of those parameters that define the structure of the analytical model and the random variable parameter vector $\boldsymbol{\theta}^A$ consists of the parameters of the random variables that are modeled in the analytical model. For the analytical model, there is no random number generator, so unlike a simulation model, for a given vector $\boldsymbol{\theta}^A$ an analytical model will produce a fixed output. For queueing network models, for example, the output vector will consist of state probabilities and/or mean performance values.

The first surrogate requirement for the conceptual analytical model states that one of the elements of the output vector $\mathbf{Y}^A$ must correspond to the simulation performance measure, $E[Y_P^S]$. We refer to this surrogate output, if it exists, as Z, since it is also the same analytical model output that serves as the ACV. By "correspond", we mean Z and $\bar{Y}_P^S$ are measures of the same phenomena as estimated by their respective models. Examples could be mean cargo loaded per day or mean aircraft throughput per hour. The difference is that $\bar{Y}_P^S$ approximates the value of the phenomena in an actual system, while Z approximates the value of $\bar{Y}_A^S$ for a valid surrogate model.

Implicit in the construction of the conceptual analytical model is that a there is a projection and transformation of elements in $\boldsymbol{\phi}^S$ and $\boldsymbol{\theta}^S$ to elements in $\boldsymbol{\phi}^A$ and $\boldsymbol{\theta}^A$. It is not expected that all elements of $\boldsymbol{\phi}^S$ and $\boldsymbol{\theta}^S$ be transformed to elements in $\boldsymbol{\phi}^A$ and $\boldsymbol{\theta}^A$. For that matter, it is not required, or expected that all elements of $\boldsymbol{\phi}^A$ and $\boldsymbol{\theta}^A$ are the direct result of a transformation of elements in $\boldsymbol{\phi}^S$ and $\boldsymbol{\theta}^S$. Obviously, the goal is to project and transform as many parameters as possible between the two models. The more transformations, the more likely that the conceptual analytical model is a valid representation of the simulation model. The actual number, or portion, of elements that must be projected for f^A to mimic f^S is indeterminate and depends not only on the models, but the particular study being performed. The best one can say is that an "adequate" number of parameters must be mapped as determined by the validation process. In previous chapters, it turns out that for the purposes of generating an ACV, adequacy can be defined in terms of the amount of variance reduction achieved. An adequate map for generating an ACV could possibly serve well for f^A to act as a surrogate of f^S. However, to meet the second surrogate requirement above, specific elements of $\boldsymbol{\phi}^S$ and $\boldsymbol{\theta}^S$ must be projected to $\boldsymbol{\phi}^A$ and $\boldsymbol{\theta}^A$. These elements are the structural or random variate parameters that are to be varied in the particular simulation study being performed. In a design of experiment context, these elements are referred to as treatments or factors. When performing a search of the experimental design space, these are the inputs to the simulation model that are varied. If there is no similar (surrogate) parameter in the analytical model, f^A can hardly be used as a surrogate of the simulation model for that specific study.

This *treatment projection and transformation* requirement can be posed in the following manner. Consider a simulation study where the measure of performance is estimated by $\bar{Y}_P^S$. Let the treatments that are to be varied in the study be represented by $\boldsymbol{\lambda}^S = (\lambda_1^S, \lambda_2^S, \ldots, \lambda_k^S)$ where $\lambda_i^S \in \left(\boldsymbol{\phi}^S, \boldsymbol{\theta}^S\right)$ for $i = 1, 2, \ldots, k$. So, for f^A to be a surrogate of f^S, there must be a function, g, that projects and transforms $\boldsymbol{\lambda}^S$ to specific elements of $\boldsymbol{\phi}^A$ and $\boldsymbol{\theta}^A$, say $\boldsymbol{\lambda}^A = (\lambda_1^A, \lambda_2^A, \ldots, \lambda_k^A)$ that corresponds directly to $\boldsymbol{\lambda}^S$. These analytical model parameters may not be exactly the same as the ones in the simulation model. A prime example is the level of aggregation for the parameter(s) modeled. In any case, the analog analytical model parameter must be adjustable in a manner that corresponds directly to the adjustments made to the parameter(s) in the simulation model. Mathematically, the treatment projection and transformation is represented by

$$g\left(\boldsymbol{\lambda}^{S}\right) = \boldsymbol{\lambda}^{A} \tag{6.4}$$

which is the second necessary condition for the conceptual analytical model to be a valid representation of the simulation model.

Figure 6.4 depicts the conceptual analytical model development and validation process in flowchart form. As shown in Figure 6.4, the process begins with a valid simulation model and a specific problem. In developing the conceptual model, the theories, assumptions, and mathematical equations used to construct it should be considered reasonable for the purposes of approximating the simulation model. This requirement is referred to face validity [45] and is depicted as one of the requirements in the flowchart decision block. Specific requirements, or conditions, for conceptual analytical model validity are twofold. The first is that there must be an output of the conceptual analytical model that approximates the performance measure of interest of the simulation model. The second requirement is that a mapping from treatments of the simulation study to the input elements of the conceptual analytical model. These requirements also appear in the flowchart decision block. Each of these requirements must be met before proceeding on to the next step of verifying the computerized analytical model. If these requirements are not met, the analyst should determine if it is cost efficient to pursue the ACV method before attempting to modify the conceptual model. After all, the goal is to reduce simulation study times, if possible. If the analyst believes it is cost efficient to proceed, the conceptual analytical model is modified until it meets the described requirements.

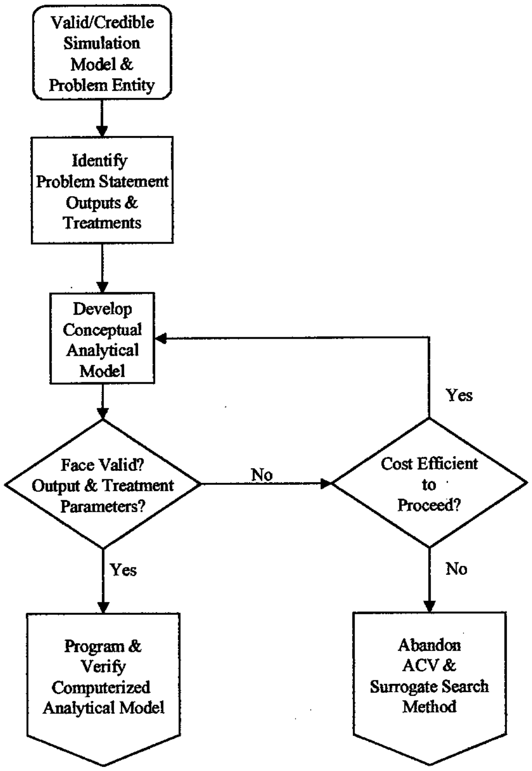
Valid/Credible Simulation Model & Problem Entity
Identify Problem Statement Outputs & Treatments
Develop Conceptual Analytical Model
Face Valid? Output & Treatment Parameters?
No
Cost Efficient to Proceed?
Yes
Yes
No
Program & Verify Computerized Analytical Model
Abandon ACV & Surrogate Search Method

Figure 6.4 Conceptual analytical model development and validation flowchart.

6.3.2 Computerized Analytical Model Verification. Verifying the computerized analytical model is essential to the surrogate search validation process. This can often be a much easier task than verifying a simulation model. For most analytical models software packages and/or well-known algorithms and mathematical formulae exist. It is a relatively straightforward task then to apply these packages and algorithms to translate the conceptual analytical model to a computerized version. Some effort, such as evaluating test problems with known answers, should be made to verify correct application of the software and/or algorithms.

6.3.3 Surrogate Search Operational Validity. Surrogate search operational validity is equivalent to achieving operational validity for a simulation model. This process examines the computerized analytical model's output to determine if it is accurate enough for its intended purpose of serving as a surrogate of the simulation model. Essentially, when performing a surrogate search, an analytical model is attempting to predict the value of the simulation model at particular points in the experimental design space. Therefore, it is logical to use the *predictive validation* technique as a means of establishing surrogate search operational validity. Recall that in the predictive validation technique, the simulation model is used to predict system behavior and then comparisons are made between the predictions and the actual observed values of the system. In this case, we make predictions with the analytical model and compare the predictions to the output of the simulation model. The comparisons are first made at each design point of an experimental design by analyzing the results of the ACV method. Then comparisons are made between the response surfaces estimated by each model over the experimental design space. If each of these comparisons is favorable, surrogate search operational validity is established. Of course, the process is iterative in that modeling errors are addressed by modifying the conceptual analytic model as necessary. We describe each of the validation steps below.

6.3.3.1 ACV Design Point Validation. We wish to perform the surrogate search operational validation in as streamlined a manner as possible to maintain our goal of reducing simulation study times. Since we are already using the ACV method to reduce variance at each of the experimental design points, the data necessary to validate the analytical model is generated at no additional cost. In a sense, we can perform "on-the-fly" validation using the ACV results. Although all design points of the simulation study must eventually be replicated and tested to validate the surrogate model, the analyst may choose to perform the initial validation tests on only a few of the design points until he/she is fairly confident in the performance of the analytical model. This reflects the iterative nature of the validation process. In this section we describe the ACV method of predictive validation and provide justification for its use.

Recall that an ACV controlled response of the performance measure of interest, $\bar{Y}_P^S(\hat{\beta})$, for n independent replications at a particular experimental design point is defined by

$$\bar{Y}_P^S(\hat{\beta}) = \bar{Y}_P^S(n) - \hat{\beta}\left(\bar{Z} - \hat{\mu}_Z\right) \tag{6.5}$$

where $\hat{\beta}$ is found using Equation (2.5). The ACV, $Z_j \in \mathbf{Y}_j^A$ for replication j is defined as

$$Z_j = f^A\left(\phi^A, \bar{\mathbf{Y}}_{A(j)}^S\right) \quad j = 1, 2, \ldots, n \tag{6.6}$$

where $\bar{\mathbf{Y}}_{A(j)}^S$ is the vector of sample means of some subset of the realized "input" stochastic processes of the simulation model observed during replication j. Therefore the stochastic processes that drive the simulation replication are correlated, by their sample means, with the input to the analytical model during the generation of an ACV. Since the variance of the ACV controlled response is given by [31]

$$Var\left[\bar{Y}_P^S(\hat{\beta})\right] = \frac{n-1}{n-3}\left(1 - R_{Y_P^S Z}^2\right) Var\left[\bar{Y}_P^S\right] \tag{6.7}$$

with

$$R_{Y_P^S Z}^2 = \frac{\sigma_{Y_P^S Z}\sigma_{Z Y_P^S}}{\sigma_{Y_P^S}^2 \sigma_Z^2} \tag{6.8}$$

the amount of variance reduction achieved is a function of the amount of covariance between $\bar{Y}_P^S$ and Z. Thus, the more the variance is reduced, the more the outputs of the two models are correlated through $\bar{\mathbf{Y}}_A^S$.

To achieve design point validity we perform the ACV method at each of the design points in the study's design of experiments (DOE). To use the ACV results for validation purposes we must change our frame of reference from variance reduction to prediction ability. Recall that if certain normality conditions are met, Lavenberg, Moeller, and Welch [30] have shown that the CV method is the same as a classical linear regression problem where $\bar{Y}_P^S$ is the dependent variable and $Z - \mu_Z$ is the independent variable. The linear regression model that is estimated as the result of computing an ACV controlled response at a particular design point is given by [31]

$$E[\bar{Y}_P^S] = \bar{Y}_P^S(\hat{\beta}) + \hat{\beta}\,(Z - \hat{\mu}_Z) \tag{6.9}$$

where the ACV controlled response, $\bar{Y}_P^S(\hat{\beta})$ is an estimated parameter of the model for the given replications (see Equations (2.17) and (2.18)) and $\hat{\mu}_Z$ is an estimate of the constant μ_Z. We show how analysis of this regression model, and its parameters, can ascertain the predictive ability of the analytical model in the neighborhood of each design point. In particular, if the linear regression model is adequate, and if $\hat{\beta} \approx 1$, and $\bar{Y}_P^S \approx \bar{Z}$ then the analytical model is an accurate predictor of the simulation model in the neighborhood of that design point. We examine each of these conditions below.

When the linear model estimated by Equation (6.9) is found to be adequate (by a statistical F-test, a large coefficient of simple determination (R^2), and/or a small MSE for example) it is appropriate to say that $E[\bar{Y}_P^S]$ can be linearly predicted by the value of Z (since $\hat{\mu}_Z$ is a constant) near the examined design point. For each replication j, the value of Z_j is computed using inputs that are sample means of the input stochastic processes of the simulation model. Hence, when both models are given approximately the same input settings, the value of the simulation model *in the neighborhood of a design point* is accurately predicted by the linear regression function of Z produced by the ACV method at that design point.

Without further analysis, we can only state that we can accurately predict the behavior of the simulation model using a different linear function of the analytical model near each of the separate experimental design points. If $\hat{\beta} \approx 1$ then we know that a unit change in Z predicts approximately a unit change in $E[\bar{Y}_P^S]$ and the value of Z is simply a linear translation of $E[\bar{Y}_P^S]$. If however,

$\bar{Y}_P^S \approx \bar{Z}$, that translation is roughly equal to zero. If each of the above conditions are met at each of the design points, the value of the analytical model approximately predicts the value of the simulation model in the neighborhood of each of the design points. It is up to the analyst to determine if the degree of predictive accuracy achieved meets the intended purpose of study and the resulting surrogate search. If it has been met, the analytical model is operationally valid near each of the design points at least.

Simple methods to perform the design point predictive validation follow. Obviously, a simple calculation can verify that $\bar{Z} \approx \bar{Y}_P^S$. By performing the ACV method, the other conditions can be quickly checked. First, $\hat{\beta}$ is estimated during the ACV method and can be examined by the analyst for closeness to the value of one. One easy method of quickly assessing the appropriateness of the suggested linear model is determining the amount of variance reduction achieved and examining a two-dimension scatter plot of $\mathbf{Z}$ versus $\bar{\mathbf{Y}}_P^S$. Since the variance of the ACV controlled estimate of Y_P^S is a function of $R^2_{\bar{Y}_P^S Z}$ and the coefficient of simple determination R^2 is the MLE of $R^2_{\bar{Y}_P^S Z}$, the amount of variance reduction achieved is a measure of model adequacy. If variance reduction achieved is relatively high, the appropriateness of the model can be further verified by examining the two-dimensional scatter plot of $\mathbf{Z}$ and $\bar{\mathbf{Y}}_P^S$. Given that the scatter plot indicates a linear relationship, combined with the variance reduction achieved, it can be safely assumed that the model appropriateness is met. Further statistical testing can be performed, as the analyst deems appropriate.

When the validation criteria fails, the analyst has a number of options. First, the cost efficiency of attempting to modify the conceptual model in order to develop a better analytical model should be considered. After a number of failed attempts, it may make more sense to abandon the ACV and surrogate search process altogether rather than spend more time on the analytical model. We also point out that if the analytical model doesn't meet all of the criteria above, it still may serve as an adequate surrogate model. For example, if $\hat{\beta} \approx c \neq 1$ where c is the same constant at all of the design points, a "new" analytical model can be constructed by simply multiplying the output of the original model by $1/c$. In a similar manner, a constant translation of the sample mean of the analytical model from the sample mean of the simulation model is solved in a similar manner. If predictive performance is poor at only a few of the design points, their lack of accuracy may not be a significant problem as long as the surrogate search is to be performed in a direction away from the poorly performing points. Results that appear unsatisfactory may also occur if the performance measure has little realized variance at some of the design points. In that case, little if any variance reduction may occur and the value of $\hat{\beta}$ may not be close to 1. However, if the sample means of each model are approximately the same, it is possible that the analytical model is a good predictor. Response surface comparisons made in the next section may indicate that the analytical model sufficiently predicts the behavior of the simulation model when considered over the entire design region. Therefore, unless the ACV design point results are highly inaccurate it is best to proceed to the surface comparisons before making a final decision on surrogate search operational validity.

6.3.3.2 Response Surface Validation. So far we have only discussed the operational validity of the analytical model near each of the experimental design points. To achieve surrogate search operational validity some means of comparing the models over the entire experimental design region is required. One approach is to predict the estimated simulation response surface with a response surface generated by the analytical model. This is an obvious choice since it is the next step in a RSM study following the replication of the simulation model at each of the initial design points. For illustrative purposes we consider only a first order surface as estimated during the initial stages of an RSM study. If the analytical model response surface accurately predicts the simulation model response surface, and the ACV design point validation is successful, the analytical model passes the predictive validation test and is therefore operationally valid as a surrogate of the simulation model in the prescribed design space.

Of course, if the ACV design point validation is met with a high degree of accuracy at each of the design points, the mean values of the analytical model and the simulation model are nearly the same at the design points. Obviously, the response surfaces generated by each of the models will then also be approximately the same and surrogate search operational validity is achieved. However, if the acceptable bounds on the ACV design point test are relatively large, the surface comparison results are less apparent. Besides that, the simulation model response surface must be generated anyway in order to proceed on with an RSM study. Since we have constructed the analytical model with an eye toward rapid computation, the computation of the analytical model response surface and then comparing it to the simulation response surface takes little additional time. There are many possible ways to compare surfaces and many possible ways of constructing the analytical model surface. This section describes one approach of constructing and comparing surfaces.

Since the surrogate methods are explored in the context of an RSM model, the approach to analytical model response surface construction and response surface comparison is suggested by RSM. The initial experimental design of an RSM model is constructed to estimate a first-order empirical model in order to define the direction of steepest ascent [12]. The gradient vector of the empirical model defines this direction. A logical approach is to construct a first order least squares model using the values of the analytical model calculated at the design points used in the simulation experiments. The gradients of the two surfaces could then be compared to see if they point in the same, or nearly the same, direction. If the gradients are very similar, this is an indication that the analytical model adequately predicts the simulation model gradient. By also comparing the response surfaces at the center of the design region (coded treatments equal zero), the relative level of the responses are assessed. Since a first order surface can be uniquely defined by its gradient and response level at the design center, favorable comparisons of both indicates that the analytical model accurately predicts the behavior of the simulation model over the entire design region and surrogate search operational validation is met. These comparisons are described below.

Consider an experimental design at an initial stage of an RSM study. Let Y_P be the performance measure of interest and let Y_P^S be a simulation model output such that $E[Y_P^S] = Y_P$. Assume that the ACV method is used to find a controlled estimate of $E[Y_P^S]$. For notational simplicity, the ACV terminology will not be used in this discussion. The reader should assume that all estimates of simulation model responses are estimated using the ACV method. Now let Θ^S represent the *coded* treatment variables that will be set in the initial experiments, where $\Theta^S = (\Theta_1^S, \Theta_2^S, \ldots, \Theta_k^S)'$ are the k different coded treatments. Assume that the initial experimental design is a 2^k factorial design so that a first order empirical model of the form

$$Y_P = b_0 + b_1\Theta_1^S + b_2\Theta_2^S + \cdots + b_k\Theta_k^S \tag{6.10}$$

can be estimated. To estimate Equation (6.10), n independent replications of the simulation model are generated at each of the 2^k design points. The inputs to the simulation model at design point i are defined in the following manner. Let $\Theta_{(i)}^S$ be the vector of coded treatment levels at design point i, $i = 1, 2, \ldots, 2^k$. These coded treatments are mapped to the treatment factor vector $\boldsymbol{\lambda}_{(i)}^S$ for design point i by

$$\Theta_{(i)}^S \longrightarrow \boldsymbol{\lambda}_{(i)}^S \tag{6.11}$$

such that

$$\boldsymbol{\lambda}_i^S \in \left(\boldsymbol{\phi}_{(i)}^S, \boldsymbol{\theta}_{(i)}^S\right) \tag{6.12}$$

Then the simulation response for the j^{th} replication at the i^{th} design point is given by

$$f^S\left(\boldsymbol{\phi}_{(i)}^S, g(\boldsymbol{\theta}_{(i)}^S)_j\right) = Y_{P(i)j}^S \tag{6.13}$$

Least squares analysis is applied to the $n \times 2^k$ responses in order to obtain the estimated parameter vector $\hat{\mathbf{b}}^S = (\hat{b}_0^S, \hat{b}_1^S, \ldots, \hat{b}_k^S)'$ so that the estimated response surface for the simulation model is given by

$$E[Y_P^S] = \hat{b}_0^S + \hat{b}_1^S \Theta_1^S + \cdots + \hat{b}_k^S \Theta_k^S \tag{6.14}$$

The usual statistical measures and tests can be performed to determine the adequacy of the estimated model. The gradient of the simulation response surface, ∇^S is given by

$$\nabla^S = (\hat{b}_1^S, \hat{b}_2^S, \ldots, \hat{b}_k^S)' \tag{6.15}$$

and the estimated response at the center of the design is $\hat{b}_0^S$.

The first-order response surface for the analytical model is estimated in a similar manner. For an analytical model that has a validated conceptual model, the response measure of interest is the same as that used for the ACV, Z, which estimates the same response as Y_P^S. To estimate the least squares response surface for the analytical model, the *nominal* response at each design point i is found by

$$f^A\left(\phi_{(i)}^A, E\left[\bar{\mathbf{Y}}_{A(i)}^S\right]\right) = Z_{(i)} \tag{6.16}$$

where

$$\boldsymbol{\lambda}_{(i)}^A \in \left(\phi_{(i)}^A, E\left[\bar{\mathbf{Y}}_{A(i)}^S\right]\right) \tag{6.17}$$

An obvious difficulty exists if some elements of $E\left[\bar{\mathbf{Y}}_{A(i)}^S\right]$ are unknown. In that case an estimate obtained from the n replications performed at that design point can be used in its place. The first-order least squares estimate of the analytical model response surface is then

$$E[Z] = \hat{b}_0^A + \hat{b}_1^A\Theta_1 + \cdots + \hat{b}_k^A\Theta_k \tag{6.18}$$

with gradient $\nabla^A = (\hat{b}_1^A, \hat{b}_2^A, \ldots, \hat{b}_k^A)'$ and the estimated response at the center of the design is $\hat{b}_0^A$. Since the first-order model is an approximation of the analytical model, not a statistical estimation, there are no statistical measures for testing the adequacy of the model. However the first-order approximation can be quickly checked by evaluating the analytical model at the center of the design and comparing the value to $\hat{b}_0^A$. If the difference between the values is small compared to the response, the approximate first-order model should be adequate for the purpose of comparing gradients.

The gradient of a first-order response surface is constant over the entire surface. Therefore if the gradients of the analytical and simulation response surfaces point in nearly the same direction, a unit change in any direction of the inputs to each model will result in a similar change in response for both models over the design region. If the directions of the two gradients are close to each other, the response level changes experienced by changing the inputs to the analytical model predict the response level changes in the simulation model. The gradient directions are compared by computing the cos of the angle, ψ, between the two vectors which is given by

$$\cos\psi = \frac{\nabla^{A'}\nabla^S}{|\nabla^A|\,|\nabla^S|} \tag{6.19}$$

where $|a|$ is the length of vector a. If $\cos\psi$ is close to one, the two vectors point in a similar direction. If an analyst wants a more objective measure, a statistical test can be performed on the hypothesis that $\nabla^S = \nabla^A$. The test is based on the $1-\alpha$ confidence cone about ∇^S. To construct the test, consider the semiplane angle $\theta_{1-\alpha}$ at the vertex of the $1-\alpha$ confidence cone between a line on the surface of the cone and the axis of the cone, which is defined by the estimated gradient vector. Box and Draper [12] show that $\sin^2 \theta_{1-\alpha}$ is given by

$$\sin^2 \theta_{1-\alpha} = \frac{(k-1) S^2_{\hat{b}^S} F_\alpha (k-1, v_b)}{\sum_{j=1}^{k} \hat{b}^S_j} \qquad (6.20)$$

where $S^2_{\hat{b}^S}$ is the estimate of the variance of any $\hat{b}^S_j$, and v_b is the number of degrees of freedom in the estimate. Note that $S^2_{\hat{b}^S_j} = S^2_{\hat{b}^S_i}$ for $i = j = 1, 2, \ldots, k$ since the coded treatment levels are used [12]. Since $\cos^2 \theta = 1 - \sin^2 \theta$, the statistical hypothesis test is posed as follows

$$\begin{aligned} H_0 : &\quad \nabla^S = \nabla^A \\ H_A : &\quad \nabla^S \neq \nabla^A \end{aligned} \qquad (6.21)$$

where the null hypothesis is rejected if

$$\cos^2 \psi = \left[\frac{\nabla^{A'} \nabla^S}{|\nabla^A| \, |\nabla^S|} \right]^2 > 1 - sin^2 \theta_{1-\alpha}$$

If it is determined that the gradients are sufficiently close, the estimated responses at the center of each response surface are also compared to determine the analytical model's predictive ability. Given sufficiently close gradients and a small relative difference (compared to the magnitude of the response) between the responses at the center of the design the first-order response surfaces for both models are relatively the same. Hence a specific input to both linear models will result in responses that are relatively the same. Given that the first-order response surface approximations of both models are appropriate, any particular set of inputs from the entire design region to the analytical model will accurately predict the simulation model response given the same inputs over the entire region. Hence, surrogate search operational validity is completely met.

As with ACV design point validation, we again must consider alternative criteria when the defined response surface validation criteria is not met. In particular, if we find that a constant translation of the sample means of the analytical model is required to meet ACV design point validation, we will need to adjust the analytical model output at each of the nominal settings in order for the design centers to be relatively close. In a similar manner, if $\hat{\beta} \approx c \neq 1$ at all of the design points, the same type of adjustment as used in the ACV design point validation process must be made here as well for the two surfaces to be oriented in the same directions. The point is that when we consider alternative criteria we are searching for ways of adjusting the analytical model so that we can use it to make accurate predictions of the simulation model output.

6.3.4 Summary. We illustrate the surrogate search operational validity process in a flowchart in Figure 6.5. The process begins with a verified computerized analytical model and proceeds through the two steps of operational validation, ACV design point validation and response surface validation. If both steps are successfully completed, the surrogate model is validated. If some of the ACV design point criteria are not, the alternate criteria described above can be evaluated to determine if the response surface validation criteria can be met. The flowchart highlights the iterative process and the concern with efficiency. If either step completely fails, the analyst must decide if it is more efficient to start the validation process over again by modifying the conceptual analytical model or to abandon surrogate search process.

Given that the surrogate search validation process is successful, the analyst is confident that the analytical model is a good surrogate for the simulation model over the investigated design region. The goal, though, is to use the analytical model as a surrogate outside of the design region. Certainly the successful validation process is the best, and only, indication that it will be capable of performing that task. The method is grounded in accepted validation techniques [45] and provides an analyst with a certain level of confidence in the analytical model. However, this process provides no guarantee. This is of course no different than validating a simulation model. Simulation studies are often performed because it is difficult, if not impossible to perform experiments on the actual system. Indeed, the system may only be a proposed system that doesn't even exist in the real

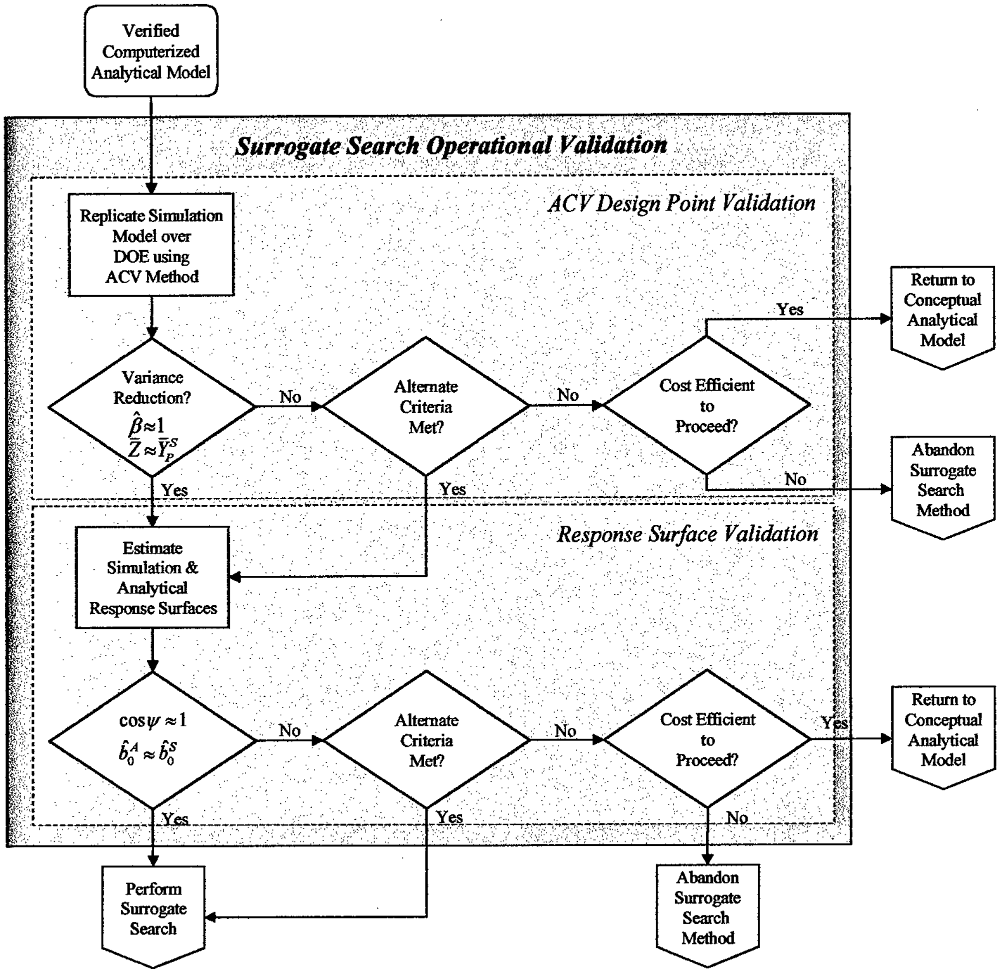
Verified Computerized Analytical Model
Surrogate Search Operational Validation
ACV Design Point Validation
Replicate Simulation Model over DOE using ACV Method
Variance Reduction?
$\hat{\beta} \approx 1$
$\bar{Z} \approx \bar{Y}_P^S$
No
Alternate Criteria Met?
No
Cost Efficient to Proceed?
Yes
Return to Conceptual Analytical Model
No
Abandon Surrogate Search Method
Yes
Yes
Response Surface Validation
Estimate Simulation & Analytical Response Surfaces
$\cos\psi \approx 1$
$\hat{b}_0^A \approx \hat{b}_0^S$
No
Alternate Criteria Met?
No
Cost Efficient to Proceed?
Yes
Return to Conceptual Analytical Model
Yes
Yes
No
Perform Surrogate Search
Abandon Surrogate Search Method

Figure 6.5 Surrogate search operational validation process flowchart.

world. Validation of the results from these types of experiments come only after the expenditure of money and time to change or develop the system as tested. For a surrogate search, we are in a much better position. In this case, the results of the surrogate search can be validated by replicating the simulation model. Furthermore, if it turns out that the surrogate search results are invalid, only the relatively short time required performing the surrogate search is lost. This "wasted time" may be more than offset by the time saved by the successful application of the ACV method of variance reduction.

In summary, by following the procedures described above, an analyst can efficiently determine the validity of using a proposed analytical model as a surrogate of a simulation model. The next step then is to actually perform the surrogate search. Surrogate search procedures are outlined in the next section.

6.4 Surrogate Search

Surrogate search methods could be used in many different situations in many different types of simulation studies. We choose to describe a surrogate search in the context of an RSM study whose goal is to find the maximum response for a given set of treatments. RSM studies can be formulated in many different ways. For the purposes of this discussion, the initial stages of a "typical" RSM study are described in the following manner. At the start of a simulation RSM study, replications are performed according to an experimental design so that a first-order empirical model can be estimated. Based on the estimated response surface, an estimated gradient is calculated. Replications are performed at design points located at multiples of the unit gradient vector positioned at the center of the original design. The replications are performed to identify the maximum response on the steepest ascent path. The maximum is used to define a new experimental design center for further experimentation. A surrogate search procedure along the path of steepest ascent is described in this section.

Using the terminology described in Section 6.3.3.2, let Y_P be the performance measure of interest and let $\bar{Y}_P^S$ be an output of simulation model f^S, such that $E[\bar{Y}_P^S] = Y_P$. Recall from Section 6.3.3.2 that the ACV method is used to estimate the simulation response. The ACV notation is not used in order to keep formulas simple. Consider a 2^k factorial experimental design where $\Theta^S = (\Theta_1^S, \Theta_2^S, \ldots, \Theta_k^S)'$ are the k different coded treatments that are varied at the 2^k different design points. By performing n independent replications of the simulation model at each design point using the appropriate inputs, the estimated gradient vector is given by

$$\nabla^S = (\hat{b}_1^S, \hat{b}_2^S, \cdots, \hat{b}_k^S)' \tag{6.22}$$

The task of defining the new experimental design points along the direction of steepest ascent is simplified by finding the unit gradient vector, v_S, by

$$v^S = \frac{\nabla^S}{|\nabla^S|} \tag{6.23}$$

In the previous paragraph it is mentioned that the ACV method is used to estimate the simulation responses. In order to achieve additional efficiency, the analyst may choose to approximate each $\mu_{Z(i)}$, the mean of the ACV at design point i, using fewer random vector replications than recommended in Chapter IV. Since this is the exploratory stage of investigation, the efficiency gained is warranted. The goal at this stage of experimentation is to find a direction to perform more experiments, not provide a final answer.

We begin by describing the steepest ascent search method for a simulation study. To perform the steepest ascent search an appropriate step size must be determined. This step size is a function of several factors, including the resolution of the uncoded treatment levels. For example, consider the case where one of the treatments corresponds to the settings of a particular machine control. If the control can only be positioned in a finite number of specific settings, the coded step size must be adjusted to account for that. Without loss of generality, assume that the coded step size is the length of the coded unit vector $|v^S| = 1$. Then the possible experimental design points can be defined as multiples of size, Δ^S, along the direction of v^S starting at the coded design center

$(0, 0, \ldots, 0)'$. Thus the steepest ascent design points, given by $\Theta^S_{sa(i)}$, are found by

$$\begin{aligned} \Theta^S_{sa(1)} &= 1\Delta^S v^S \\ \Theta^S_{sa(2)} &= 2\Delta^S v^S \\ \vdots & \quad \vdots \\ \Theta^S_{sa(i)} &= i\Delta^S v^S \\ \vdots & \quad \vdots \end{aligned} \tag{6.24}$$

where the total number of design points is determined by the results of the experiments [12].

At this point, the analyst is faced with some important decisions. What is an appropriate value for Δ^S? Given that value for Δ^S, should experiments be conducted at each integer multiple of Δ^S or can some of the design points be skipped? If the analyst is conservative and the actual maximum is relatively far from the design center, many "uninteresting" time-consuming replications will be performed. Or the opposite might occur, where the initial step is chosen as very large and the maximum is very close to the design center. Prior knowledge of the system or expert opinion can guide the analyst in choosing the appropriate step size. This is exactly what the analytical model can provide when used as a surrogate to the simulation model along the direction of steepest ascent.

To perform a surrogate search, the analytical model is evaluated at a set of design points along the path of steepest ascent in order to identify the maximum response. Given that surrogate search validation is achieved, it is likely that the treatment levels at the design point that corresponds to the maximum analytical model response will be close to the treatment levels that will produce the maximum response in the simulation model. Conducting a number of replications at or near that design point can validate this. Numerous methods of performing the surrogate search are possible. We begin by describing a method that mimics the simulation steepest ascent method described above. Some alternate methods are sketched out at the end of the section.

Since the analytical model can normally be evaluated in virtually no time at all when compared to performing multiple replications of the simulation model, the step size for the surrogate search should be relatively small. The actual size chosen can be affected by the space of permissible values of each of the treatment levels. This limitation can occur separately in each model for different treatments, or in both models for the same treatment. When the limitation is the same in both models, the analyst simply adjusts the step size as before. However, if a treatment can take on only a finite number of values in the simulation model, but can take on more values in the analytical model, there is little to gain by evaluating those additional treatment levels. After all, the resultant response at those levels can never be validated by the simulation model. On the other hand, there may be treatments that have a fewer number of permissible values in the analytical model, than the simulation model. In that case, the analyst has no choice but to adjust the surrogate search step size based on the treatment level limitation. Nonetheless, the surrogate search still provides the analyst information about how the simulation model might act along the path of steepest ascent. Therefore, based on these treatment resolution factors, the analyst should choose a surrogate search step size, Δ^A, as small as practicable.

Given a step size of Δ^A, the surrogate search design points, $\Theta^A_{ss(i)}$, are given by

$$\begin{aligned} \Theta^A_{ss(1)} &= 1\Delta^A \upsilon^S \\ \Theta^A_{ss(2)} &= 2\Delta^A \upsilon^S \\ \vdots \quad & \quad \vdots \\ \Theta^A_{ss(i)} &= i\Delta^A \upsilon^S \\ \vdots \quad & \quad \vdots \end{aligned} \tag{6.25}$$

where the total number of design points is determined by the results of the surrogate search. The coded design points are converted to the uncoded treatment levels given by

$$\boldsymbol{\lambda}^{A}_{ss(i)} \in \left(\boldsymbol{\phi}^{A}_{ss(i)}, E\left[\bar{\mathbf{Y}}^{S}_{ss(i)}\right],\right) \tag{6.26}$$

where $E\left[\bar{\mathbf{Y}}^{S}_{ss(i)}\right]$ is the analytical model input vector of expected values of the means of the random processes of the simulation model and $\boldsymbol{\phi}^{A}_{ss(i)}$ is the vector of structural parameters at each surrogate search design point $ss(i)$. Given the design points, the surrogate search is conducted by evaluating the analytic model at each of these points by

$$\begin{aligned}
f^{A}\left(\boldsymbol{\phi}^{A}_{ss(1)}, E\left[\bar{\mathbf{Y}}^{S}_{ss(1)}\right],\right) &= Z_{ss(1)} \\
f^{A}\left(\boldsymbol{\phi}^{A}_{ss(2)}, E\left[\bar{\mathbf{Y}}^{S}_{ss(2)}\right],\right) &= Z_{ss(2)} \\
\vdots \qquad\qquad &\qquad \vdots \\
f^{A}\left(\boldsymbol{\phi}^{A}_{ss(i)}, E\left[\bar{\mathbf{Y}}^{S}_{ss(i)}\right],\right) &= Z_{ss(i)} \\
\vdots \qquad\qquad &\qquad \vdots
\end{aligned} \tag{6.27}$$

As previously mentioned, some of the values of $E\left[\bar{\mathbf{Y}}^{S}_{ss(i)}\right]$, may not be known. In that case, estimates based on previous simulation replications can be used. If the values are subject to unknown changes as the design points move out the path of steepest ascent, estimates can be made by constructing an estimated response surface using the data obtained in the original experiments.

The final step in the surrogate search method is to validate the analytical model results using the simulation model. Assume that the maximum response of the analytical model on the path of steepest ascent is given by Z_m which corresponds to surrogate search design point $ss(m)$. A simple validation method is accomplished by performing simulation replications at $ss(m)$ and two sets of replications at design points that are a distance of $\pm\Delta_m$ from $ss(m)$. Any number of other validation methods are possible based on the particular circumstances of the study. If the simulation model doesn't validate the surrogate search results, the analyst simply designs a new search based on the simulation results and his/her best judgement.

The ability to compute the analytical model rapidly provides an analyst with many options for conducting a surrogate search. Although, we have developed a procedure for the surrogate search method along the path of steepest ascent, there is no reason that the analyst should feel constrained to that path only. In fact, there are situations when it is advantageous to conduct the surrogate search differently. For example, if the resolution of one, or several, of the treatments don't allow for small steps it may be better to modify the procedure. In that case, searches with sufficiently small step sizes can be performed by holding the problem treatment constant and varying the other treatments. Another strategy might be to perform additional searches on paths other than the steepest ascent. The point is that the computational advantage of the analytical model provides the analyst with a means of exploring the experimental region more thoroughly than possible with simulation model alone. The number of possible surrogate searches is only limited by the imagination of the analyst.

6.5 Summary

We have presented a new method of using an external analytical model to reduce simulation study times by employing the surrogate search procedure. The method is a logical extension of the ACV method of variance reduction. We have adapted the simulation model verification and validation method as a means of justifying the surrogate search method for a specific simulation study. Given a validated surrogate model, searches are performed by the analytical model in order to identify promising points to perform simulation replications. In the next chapter, we demonstrate the effectiveness of the procedure by examining two different simulation models and problems.

VII. Application of Surrogate Search Method

7.1 Overview

We demonstrate the effectiveness of the surrogate search method developed in Chapter VI on two different simulation models. The first demonstration consists of a simple two factor RSM study using the Psuedo-BRACE simulation model from Chapter V. We present this simple study to demonstrate the basic application of the surrogate search method. We follow that with a RSM simulation study using the USAF HQ/AMC Airlift Flow Model (AFM) simulation model. In this case, we focus on a "real-world" size problem using an actual validated and credible simulation model. Several "non-standard" surrogate search issues, and their resolution, are examined in this case study.

7.2 Psuedo-BRACE RSM Study

A simple RSM simulation study is conducted using Pseudo-BRACE and the analytical model presented in Chapter V to demonstrate the surrogate search method. We assume that Psuedo-BRACE is a valid/credible simulation model for the purposes of this example. We begin by describing the problem and the resulting RSM study. We then describe the settings and output statistics used in Pseudo-BRACE to conduct the study. This if followed by a description of the surrogate search validation process including results from each of the steps. Finally a surrogate search is performed using the validated analytical model. The reader should pay particular attention to how the initial steps of the RSM study are performed as a result of the surrogate search validation procedure.

7.2.1 Study Description. The goal of the RSM study is to find the maximum steady state mean for the amount of cargo, C, that can be uploaded at a fictional airfield every 24 hours. Two types of fictional aircraft (C-A and C-B) are used to move the cargo, each with a different cargo capacity. Both aircraft also have different fuel load requirements and different unscheduled maintenance probability distributions. The treatments that are varied for the study are the proportions of each type of aircraft and the overall arrival rate of aircraft to the airfield. The two treatments are defined in the following manner. Let x_1 be the proportion of arriving aircraft that are type C-A aircraft, where $1/6 \leq x_1 \leq 1$ are the possible values of x_1. Then $1 - x_1$ is the proportion of arriving aircraft that are C-B's. Let x_2 represent the rate at which aircraft of any type arrive at the airfield, in aircraft per hour.

Since the airfield has a limited number of parking spots, it is expected that an increasing number of aircraft will divert as cargo up-load is maximized. It is reasonable to expect that AMC planners would want to keep the number of diverting aircraft below a certain minimum. Therefore the maximization problem is constrained by a minimum number of aircraft diverts per aircraft arrival. This constraint is expressed as the probability that an arriving aircraft will divert. Assume that AMC wants $P(\text{Divert})$ to be less than 0.05.

The RSM maximization problem can then be posed as

$$\begin{aligned} &\max \ C \\ &\text{s.t. } P(\text{Divert}) \leq 0.05 \end{aligned} \tag{7.1}$$

The first step is to construct the initial experimental design to estimate a first order empirical model using Pseudo-Brace. Assume that the AMC planners suggest that a suitable design center is $x_1 = 1/2$ and $x_2 = 1.25$ aircraft per hour. Using that as the center, a 2^2 factorial design is constructed in the following manner. Based on the planners' suggestions the high and low levels for x_1 are $x_1^H = 2/3$ and $x_1^L = 1/3$. For treatment two, the high and low levels are $x_2^H = 1.5$ and $x_2^L = 1.0$ aircraft per hour. The initial experimental design is listed in Table 7.1. The levels of the coded treatment variables, $\Theta = (\Theta_1, \Theta_2)'$ are found using the following formulas

$$\Theta_1^j = \frac{x_1^j - 1/2}{1/6} \quad j = H, L \tag{7.2}$$

$$\Theta_2^j = \frac{x_2^j - 1.25}{0.25} \quad j = H, L \tag{7.3}$$

The coded experimental design appears in Table 7.2.

Table 7.1 Initial RSM study 2^2 factorial design.

	Uncoded	
Design	Treatment Level	
Point	x_1	x_2
1	1/3	1.0
2	2/3	1.0
3	1/3	1.5
4	2/3	1.5

Table 7.2 Initial RSM study 2^2 factorial design.

	Coded	
Design	Treatment Level	
Point	Θ_1	Θ_2
1	-1	-1
2	+1	-1
3	-1	+1
4	+1	+1

7.2.2 Pseudo-BRACE Settings. We begin by describing the airfield resources and aircraft settings within Pseudo-BRACE. The airfield being simulated is essentially the same as the one studied in Chapter V, with one difference. There are nine K-loaders at this airfield versus the eight assigned to the airfield in chapter V. The airfield resources are listed in Table 7.3. Other network settings include the length of the standard ground time, 2.25 hours, and the length of time an aircraft will wait before diverting, which is 2 hours. Recall that an aircraft without hazardous cargo will be parked at a spot with a fuel hydrant pit, if one is available. If not, it will be parked first at a non-hazardous parking spot and then a hazardous parking spot. The probability that an arriving aircraft will up-load hazardous cargo is 0.10.

Table 7.3 RSM study airfield resources.

Resource	Number
Total Parking spots	12
Hydrant parking spots	4
Hazardous parking spots	4
Fuel Trucks	8
Fuel fill stand	1
K-Loaders	9
Forklifts	19
Loading docks	10

The aircraft attributes are provided in Table 7.4. Based on the fuel load requirements and fuel truck capacities, fuel trucks will have to make 2 trips to refuel a C-A aircraft and 3 trips to refuel a C-B aircraft. The cargo up-load requirement and average pallet cargo weight results in a total of 12 pallets for every C-A up-load and 15 pallets for every C-B up-load. This translates to 3 K-loaders (K-loader capacity equals 5 pallets) for every aircraft. The aircraft unscheduled maintenance category probabilities are recorded in Table 7.5. Within each category, the unscheduled maintenance time is uniformly distributed. Scheduled maintenance (performed concurrently with unscheduled maintenance) is fixed at 30 minutes and LOX servicing is fixed at 9 minutes.

Table 7.4 RSM study aircraft parameters.

Type	Probability	Fuel load (gals)	Cargo load (tons)	Fuel receive rate (gpm)
C-A	x_1	10000	26.0	450.0
C-B	$1 - x_1$	15000	32.6	450.0

Table 7.5 RSM study unscheduled maintenance probabilities.

	Time category (hours)							
Type	0-4	4-8	8-12	12-16	16-24	24-48	48-72	None
C-A	0.000	0.010	0.000	0.000	0.010	0.000	0.010	0.930
C-B	0.043	0.057	0.029	0.036	0.021	0.021	0.007	0.786

At each design point of the initial study shown in Table 7.2, 20 replications of the Pseudo-BRACE model are generated. Aircraft arrive to the airfield via a simulated Poisson process with mean arrival rate equal to the appropriate value of x_2 every hour. One thousand aircraft arrive at the airfield during each simulation replication. To eliminate the effects of initial transient bias, statistics are gathered on the last 900 arrivals. The reader should interpret all definitions of the following statistics to implicitly include these truncations. The steady state mean of cargo up-loaded every 24 hours, C, is estimated in the following manner. Let c_{ij} be the amount of cargo up-loaded by the i^{th} aircraft to depart the airfield during simulation replication j. Further let h_j represent the total time simulated in hours and d_j be the number of aircraft that depart the airfield (non-diverting aircraft) during replication j. Then the mean amount of cargo up-loaded every 24 hours for replication j is given by

$$\gamma_j^S = \frac{24}{h_j} \sum_{i=1}^{d_j} c_{ij} \quad j = 1, 2, \ldots, 20 \tag{7.4}$$

so that the mean value for cargo up-loaded every 24 hours is given by

$$\hat{C}^S = \frac{1}{20} \sum_{j=1}^{20} \gamma_j^S \tag{7.5}$$

To estimate the probability that an arriving aircraft will divert, consider the indicator variable, I_{ij}, defined as

$$I_{ij} = \begin{cases} 1 & \text{if } i^{th} \text{ arriving aircraft for rep } j \text{ diverts} \\ 0 & \text{otherwise} \end{cases} \tag{7.6}$$

Then the probability of diverting, δ_j^S, for replication j is given by

$$\delta_j^S = \frac{1}{d_j} \sum_{i=1}^{d_j} I_{ij} \quad j = 1, 2, \ldots, 20 \tag{7.7}$$

so that $P(\text{Divert})$ is estimated by

$$\bar{\delta}^S = \frac{1}{20}\sum_{j=1}^{20}\delta_j^S \tag{7.8}$$

Each of these responses are calculated at each design point in order to estimate the first order empirical model.

7.2.3 Surrogate Search Validation / Initial RSM Results. We now focus on the surrogate search validation process and the initial results of the RSM study. We begin by establishing conceptual analytical model validity. Then surrogate search operational validity is established during the initial stage of the RSM study.

7.2.3.1 Conceptual Analytical Model Validity. Given a valid/credible simulation model and a problem statement with known outputs and treatments, there are essentially two requirements for conceptual analytical model validity. The first requirement is model face validity and the second requires the analytical model to have outputs and treatment parameter inputs that correspond to the outputs and treatments of the simulation model. See Figure 7.1 for a flowchart of the process. Thus far, we have described the credible simulation model (Psuedo-BRACE) and identified the problem statement, output performance measures, and input treatments. We address the two conceptual analytical model validity requirements below.

We begin by examining conceptual analytical model face validity. The proposed conceptual analytical model is the single class closed queueing network model described in Chapter V and depicted in Figure 5.3, modified to accommodate two classes (chains) of aircraft. As before, we solve the model using the MVA algorithm and the fork-join node approximation. To modify the model for two classes, the service disciplines at the stations representing cargo up-load (station 4) and refueling (station 6 and 7) are changed from first-come first-served (FCFS) to processor sharing (PS) service disciplines. This is necessary since all classes of customer at FCFS stations must have

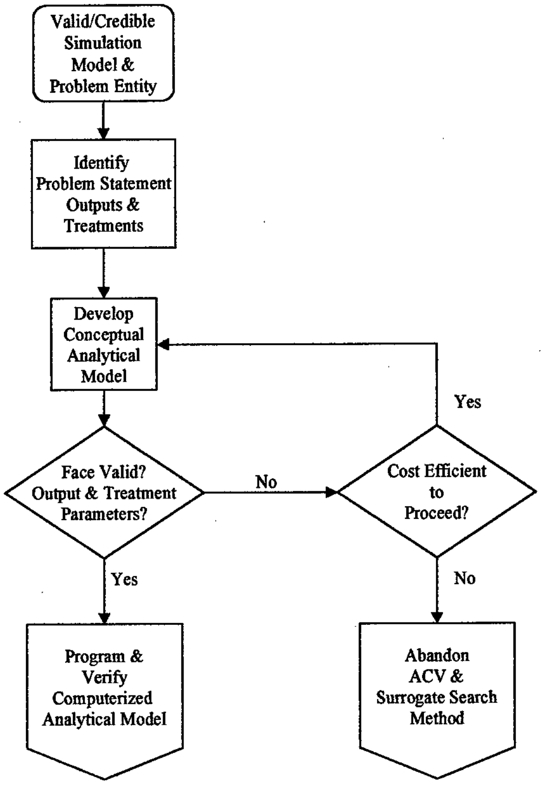
Valid/Credible Simulation Model & Problem Entity
Identify Problem Statement Outputs & Treatments
Develop Conceptual Analytical Model
Face Valid? Output & Treatment Parameters?
No
Cost Efficient to Proceed?
Yes
Yes
No
Program & Verify Computerized Analytical Model
Abandon ACV & Surrogate Search Method

Figure 7.1 Conceptual analytical model validation flowchart.

the same mean service time in order to be solved by the MVA algorithm [8,42]. However, different classes of customers at PS stations may have different mean service times that correspond to the class of the customer. PS stations assume that the station servers serve all customers in the station in parallel, each customer receiving an equal share of the service. This type of service discipline is also referred to as time division.

The MVA algorithm is simply modified to account for more than one class of customer [13,16]. Essentially an additional loop is added to the algorithm to account for each different class of customer. As with the single class MVA algorithm, the response time for each class of customer is calculated for population size $\mathbf{N} = (N_1, N_2, \ldots, N_r)'$, where r is the number of classes in the network, based on the mean queue length when one fewer customer is in the system. The cycle time for each class of customer is then calculated so that throughput and utilization for each class of customer can be found. These values are then used to find the mean queue length of each customer class at each station for customer population N. The fork-join network response times are also found based on the population of each customer class in the same manner as for one class.

The nominal service times for each class of customer and the number of servers at each design point are based on Pseudo-BRACE. These values are listed in Table 7.6. The number of cargo servers is based on the total number of K-loaders in Pseudo-BRACE divided by the number of K-loaders required by each aircraft. The proportion of aircraft refueled by hydrant system, p_H, is also required in order to evaluate the analytical model. The nominal value of this probability is not known until the simulation model is replicated. An estimate of $p_H = 1/3$ can be used based on dividing the number of parking spots with hydrant systems, 4, by the total number of parking spots, 12. Or the observed sample mean of this value from the simulation replications can be used instead.

Table 7.6 MVA model RSM study settings.

			Mean Service Times (hours)	
Station	Discipline	Servers	C-A	C-B
0	FCFS	1	$1/x_2$	$1/x_2$
1	FCFS	1	0.033	0.033
2	Delay	Inf	0.167	0.167
3	Delay	Inf	1.505	3.372
4	PS	3	0.936	1.117
5	Delay	Inf	0.160	0.160
6	PS	1	0.410	0.596
7	PS	8	1.683	2.762
8	Delay	Inf	0.167	0.167
9	FCFS	1	0.033	0.033

Based on the results of the ACV application for the single class version of this model in Chapter V, it is reasonable to expect the proposed conceptual model to adequately approximate Psuedo-BRACE for the problem described, subject to operational testing. There is a certain amount of uncertainty concerning the use of the PS service discipline at the cargo up-load and refueling stations. Obviously the service at those queues in Pseudo-BRACE do not operate in this fashion. However, the goal is to find an approximate solution that can provide results based on the different aircraft parameters. This assumption allows the model to account for the different service time parameters of the two different types of aircraft. Despite the uncertainty, it is reasonable to expect that the results from these PS queues will approximate the behavior Psuedo-BRACE queues. Therefore, at this point of the study, we believe the conceptual analytical model meets the requirements of face validity.

We now determine if the conceptual analytical model provides for the necessary output and treatment mappings. First of all, the analytical model must produce mean responses for the cargo up-loaded in 24 hours and the probability of diverting. These responses can be calculated as functions of the calculated throughput at station 0. Let $\boldsymbol{\lambda} = (\lambda_1, \lambda_2)'$ be the throughput (aircraft per hour)at station 0 for aircraft classes C-A and C-B respectively. Using the cargo capacity for each type of aircraft given in Table 7.4 as 26 and 32.6 tons for C-A and C-B aircraft, the mean number of tons of cargo up-loaded every 24 hours is given by

$$\gamma^A = 24[26 \ 32.6]\boldsymbol{\lambda} \tag{7.9}$$

For an open capacitated queue, aircraft will divert when they arrive to an airfield that is at capacity. (Note, this is different than Pseudo-BRACE, where aircraft won't divert until they have waited a total of 2 hours for a parking space.) The equivalent closed queueing network is at capacity when station 0 is idle. The probability that station 0 is idle is given by $1 - U_0$, where U_0 is the station 0 server utilization (expected number of servers busy). U_0 is found by

$$U_0 = s_{01}\mathbf{1}'\boldsymbol{\lambda} = 1/x_1\mathbf{1}'\boldsymbol{\lambda} \tag{7.10}$$

where $s_{01} = s_{02}$ is the mean service time at station 0 of type C-A and C-B aircraft respectively. Thus the probability that an arriving aircraft will divert is calculated by the analytical model as

$$\delta^A = 1 - 1/x_1 \mathbf{1}' \boldsymbol{\lambda} \tag{7.11}$$

To find γ^A and δ^A for the purposes of generating an ACV, the observed sample means of the aircraft arrival rate, unscheduled maintenance, cargo up-load time, and aircraft refueling by hydrant, for every simulation replication, are used as inputs to the analytical model. Similarly, a surrogate search for these output performance measures can be performed by providing the analytical model with the appropriate input parameters.

Now we describe how the two treatment levels are adjusted in the conceptual analytical model. Since the Pseudo-BRACE airfield has a total of 12 parking spots, the closed analytical model has a total population of 12 aircraft. Thus, the aircraft population in the analytical model is represented by $\mathbf{N}^A = (N_1^A, N_2^A)'$ such that $N_1^A + N_2^A = 12$. Let N_1^A and N_2^A represent the number of C-A aircraft and C-B aircraft in the system respectively. Then the treatment corresponding to the proportion of aircraft is adjusted in the analytical model by changing the totals of aircraft for each class. For example, for $x_1 = 1/3$, $N_1^A = 4$ and $N_2^A = 8$. This is an example of the situation described above when the resolution of the two models is not the same. Nonetheless, the first treatment level can be adjusted in the analytical model. To adjust the aircraft arrival rate, the mean service time for both types of aircraft at the "arrival" station 0 is adjusted appropriately. The mean service time at station 0 for this closed system corresponds to the reciprocal of the arrival rate for an equivalent open capacitated network. Hence for $x_2 = 1.5$, let $s_{01} = s_{02} = 1/1.5 = 2/3$, where s_{0r} is the mean service time at station 0 for customer class $r = 1, 2$. In this case, the treatment levels have the same resolution (infinite) for both models. Thus, the initial experimental design settings for the analytical model is shown in Table 7.7.

Table 7.7 Analytical model settings for 2^2 factorial design.

Design	Uncoded Treatment Level	
Point	**N**	s_{0r}
1	$[4\ 8]'$	1.0
2	$[8\ 4]'$	1.0
3	$[4\ 8]'$	1.5
4	$[8\ 4]'$	1.5

We have now shown that conceptual analytical model validity is achieved with the proposed MVA model. The next step is to perform the initial design of experiment using the ACV method in order to assess surrogate search operational validity and complete the initial stage of the RSM study. We skip the computerized analytical model verification process other than to say the computer implementation of the conceptual model was verified as correctly applied.

7.2.3.2 Surrogate Search Operational Validity. Surrogate search operational validity is a two step process that is accomplished during the first two steps of the RSM study. See Figure 7.2. First, we replicate the simulation model to estimate the output performance measures using the ACV method. The ACV results are analyzed at each design point to assess the first level of operational validity. This is referred to as ACV design point validation. Secondly response surfaces, and gradients, are estimated using results from both models. The response surfaces are then compared to judge the operational validity of the analytical model. When each of these requirements are met, the analytical model meets surrogate search operational validity.

To assess ACV design point validity and to complete the RSM study, we perform 20 replications of the Pseudo-BRACE model at each of the 2^2 design points. The ACV method is used to provide controlled estimates of the mean amount of tons of cargo up-loaded every 24 hours, $\hat{C}^S$, and the estimated probability that an aircraft diverts, $\bar{\delta}^S$. The ACV for cargo up-loaded is γ^A and for the probability of diverting, it is δ^A. It turns out that the observed number of aircraft diverting at each design point in the initial study is so low that meaningful comparisons for the probability

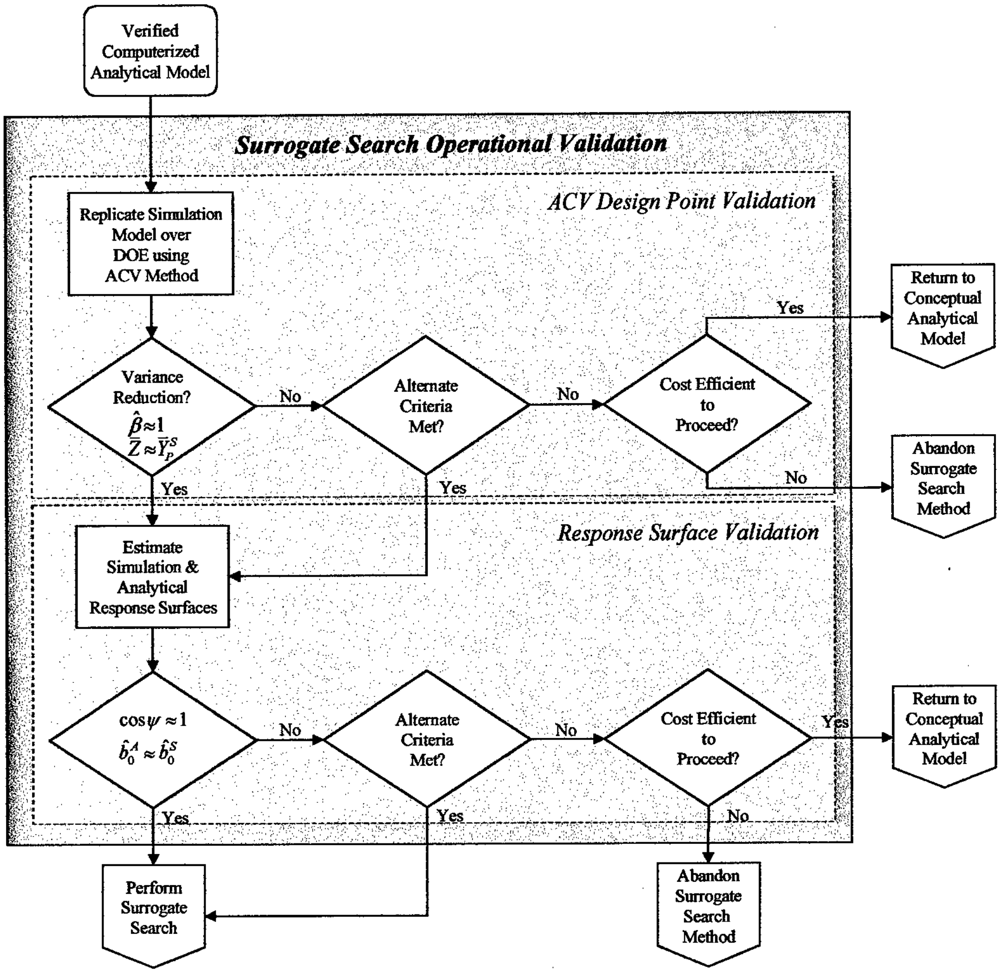
Verified Computerized Analytical Model
Surrogate Search Operational Validation
ACV Design Point Validation
Replicate Simulation Model over DOE using ACV Method
Variance Reduction?
$\hat{\beta} \approx 1$
$\bar{Z} \approx \bar{Y}_P^S$
No
Alternate Criteria Met?
No
Cost Efficient to Proceed?
Yes
Return to Conceptual Analytical Model
No
Abandon Surrogate Search Method
Yes
Yes
Response Surface Validation
Estimate Simulation & Analytical Response Surfaces
$\cos\psi \approx 1$
$\hat{b}_0^A \approx \hat{b}_0^S$
No
Alternate Criteria Met?
No
Cost Efficient to Proceed?
Yes
Return to Conceptual Analytical Model
Yes
Yes
No
Perform Surrogate Search
Abandon Surrogate Search Method

Figure 7.2 Surrogate search operational validation flowchart.

of diverting are not useful. Since the probability of diverting as calculated by the analytical model is a linear function of aircraft throughput, the ACV method is used to find controlled estimates of total aircraft throughput at each design point. Let the estimated mean aircraft throughput for Pseudo-BRACE be given by $\hat{L}^S$. The ACV for throughput is represented by $\boldsymbol{\lambda}$. These results are also compared for the purposes of judging ACV design point validity using the following criteria:

1. ACV linear regression model is appropriate

    (a) "Significant" variance reduction

(b) Linear scatter plot

2. $\hat{\beta} \approx 1.0$

3. $\bar{Y}^S \approx \bar{Z}$

The variance reduction achieved, β, $\bar{Y}_P^S$, and $\bar{Z}$ for each response are provided in Table 7.8. At design point 2, an ACV controlled response was not calculated for probability of diverting since no aircraft diverted for any of the 20 replications. Note that the mean of each ACV is estimated using only 100 random vectors at each design point. As mentioned previously, since the goal is to find a gradient, not the final answer, the time saved at this level of investigation will pay off.

Table 7.8 ACV results at all design points.

Response	Variance Reduction (%)	$\hat{\beta}$	$\bar{Y}_P^S$	$\bar{Z}$
Design Point 1 (LL)				
Cargo	85.20	0.928	724.4	729.9
P(Divert)	8.07	0.414	0.00	0.00
Throughput	87.83	0.924	0.99	1.00
Design Point 2 (HL)				
Cargo	86.61	0.990	670.9	677.2
P(Divert)	~	~	0.00	0.00
Throughput	86.31	0.947	0.99	1.00
Design Point 3 (LH)				
Cargo	63.76	0.915	1073.6	1085.0
P(Divert)	13.15	0.618	0.00	0.01
Throughput	66.23	0.910	1.47	1.49
Design Point 4 (HH)				
Cargo	68.97	1.038	996.7	1009.0
P(Divert)	-2.16	-0.095	0.00	0.01
Throughput	66.69	0.990	1.47	1.50

Two dimensional scatter plots for aircraft throughput at each design point are provided in Figure 7.3. Since the scatter plots for cargo up-loaded are essentially the same as the throughput plots, they are not provided. The scatter plots for P(Divert) are not provided since few diverts actually occurred.

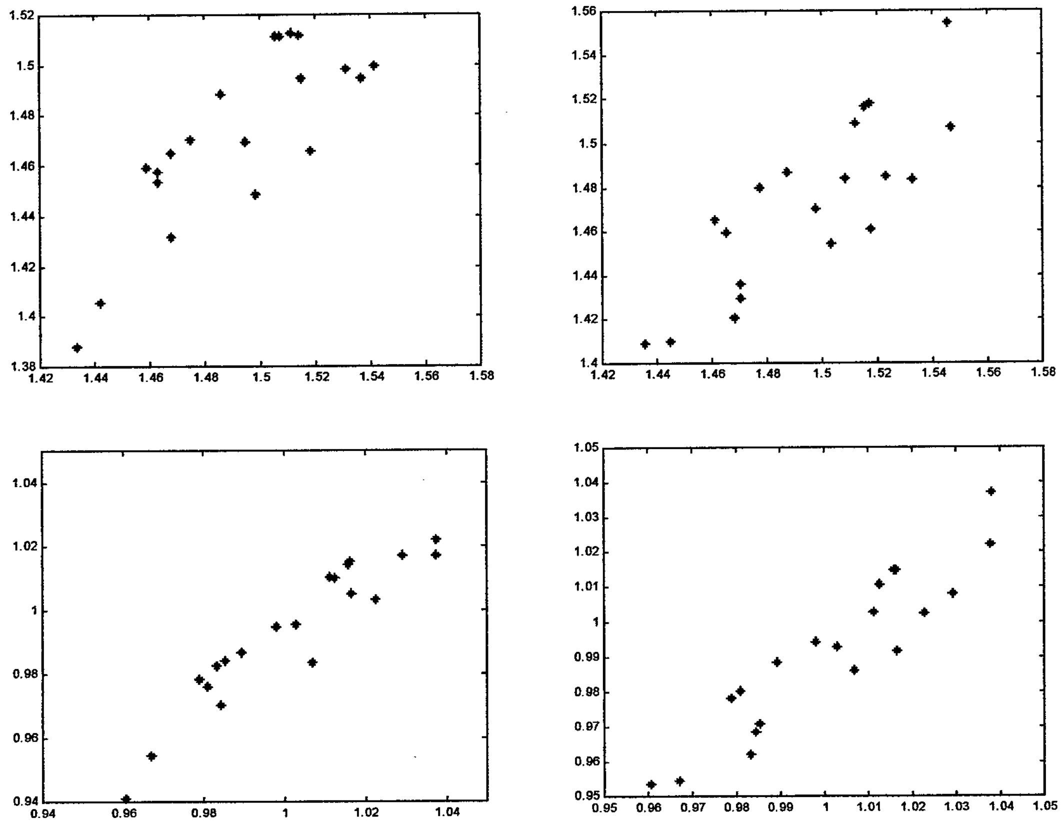

Figure 7.3 Two way scatter plots for aircraft throughput (aircraft/hr).

Based on the results presented in Table 7.8 and the scatter plots, the ACV design point validation criteria are met for the cargo up-load and throughput responses. The ACV results for probability of diverting are essentially disregarded due to the small number of diverts observed. However, the analytical model did predict small diverting probabilities at each design point. Additionally, the probability of diverting is a linear function of throughput in the analytical model, and the results for throughput are satisfactory. Based on those facts, ACV design point validity is met.

The simulation and analytical model response surfaces are estimated in order to verify the validity criteria at the response surface level and to identify the steepest ascent gradient. The estimated coefficients for the simulation and analytical models are provided in Tables 7.9 and 7.10. The estimated variance for the estimated simulation coefficients are also provided. The simulation response surfaces are based on ACV controlled responses for cargo up-loaded and throughput. For the probability of diverting response, the estimates are based on the uncontrolled responses since the ACV method could not be applied at design point 3.

Table 7.9 Simulation response surface parameter estimates.

Response	$\hat{b}_0^S$	$\hat{b}_1^S$	$\hat{b}_2^S$	$\hat{\sigma}^2_{\hat{b}}$
Cargo	866.39	-32.60	168.74	2.0882
P(Divert)	0.000783	-0.00061	0.000757	3.15×10^{-6}
Throughput	1.23	-3.50×10^{-5}	0.24	3.74×10^{-8}

Table 7.10 Analytical response surface parameter estimates.

Response	$\hat{b}_0^A$	$\hat{b}_1^A$	$\hat{b}_2^A$
Cargo	862.6426	-29.2648	165.4341
P(Divert)	0.0075	-0.0028	0.0066
Throughput	1.24	0.0042	0.24

The criteria for response surface validity are that the gradient vectors of each surface be approximately equal and that the response at the center of each surface be relatively the same value. The gradient vectors, ∇^S and ∇^A, are compared by finding $\cos\psi$, where ψ is the angle formed between the two gradients. If $\cos\psi \approx 1$ the gradient vectors point in similar directions. The gradients are also compared by performing a statistical test. The test determines if ∇^A is contained in the $1-\alpha$ confidence cone about the ∇^S. The test is reported by listing the angle that defines the 95% confidence cone ($\psi_{0.05}$). This value can then be compared to ψ. The values of ψ, $\cos\psi$, $\psi_{0.05}$ and the responses at the center of each response surface are provided in Table 7.11 for all three responses.

Table 7.11 Response surface condition results.

Response	ψ	$\cos\psi$	$\psi_{0.05}$	$\hat{b}_{S(0)}$	$\hat{b}_{A(0)}$
Cargo	0.98	0.99	0.96	866.4	862.6
P(Divert)	15.79	0.96	23.33	0.00	0.00
Throughput	1.00	0.99	0.84	1.23	1.24

The results of the comparisons listed in Table 7.11 indicate that the response surfaces compare favorably. Although ψ for cargo up-load and throughput is larger than the respective values of $\psi_{0.05}$, the angles are still very small as reflected by the $\cos\psi$ values. However, the $\psi_{0.05}$ values for these two responses are very small due to the small estimated variances of the gradient vector components. The value of ψ for probability of diverting is larger than the other two angles, but the cos is very close to one and ∇^A is contained in the 95% confidence cone about ∇^S. Therefore it is safe to conclude that the directions of the simulation and analytical model gradient vectors point in nearly the same direction for all responses. The responses at the center of each design are also very close for the response surfaces generated by both models. Hence, the response surfaces generated by both models are indeed very similar. Based on the reported experimental results, the conditions for response surface validity, are met. Both elements of surrogate search operational validation have been met, providing us with a validated surrogate search model.

7.2.4 Surrogate Search Results. Given a validated surrogate search model, we conduct the surrogate search. Since the amount of cargo up-loaded is the response that is to be maximized, the gradient based on the cargo up-load surface is used for the steepest ascent search. As mentioned previously, the gradient estimated via the simulation response surface will be used to determine the search direction. The unit gradient vector, v^S for cargo up-load is given by

$$v^S = \begin{bmatrix} -0.1897 \\ 0.9818 \end{bmatrix} \tag{7.12}$$

Using the surrogate search method outlined in Section 6.4, the next step is to determine an appropriate step size for the surrogate search. The resolution of the aircraft proportion treatment level within the analytical model and v^S determines the possible step sizes. The center of the design region corresponds to $x_1 = 1/2 \rightarrow \mathbf{N}^A = (6,6)'$. The direction of steepest ascent points in a negative direction for aircraft proportion. Therefore each surrogate search step from the design center will reduce the number of C-A aircraft in the MVA model network by one. This can continue until $N_1^A = 2$ since the aircraft proportions are constrained by $1/6 \leq x_1 \leq 1$. At that point, the search can continue with the same step sizes for x_2 holding $x_1 = 1/6$. Reducing the number of C-A aircraft by one corresponds to changing the coded treatment level by -0.5. The coded step size is then found by solving

$$\Delta^S = \frac{-0.5}{v_1^S} = 2.66 \tag{7.13}$$

Based on this step size, the coded and uncoded values of the treatment levels for the surrogate steepest ascent search are provided in Table 7.12. Note that the first step is skipped since it is inside the design region.

Table 7.12 Surrogate search steps.

	Treatment Levels			
	Coded		Uncoded	
Step	θ_1	θ_2	N	$1/s_{0r}$
0	0.0	0.0	$[6\ 6]'$	1.00
1	-1.0	5.18	$[4\ 8]'$	2.29
2	-1.5	7.76	$[3\ 9]'$	2.94
3	-2.0	10.35	$[2\ 10]'$	3.59
4	-2.0	13.00	$[2\ 10]'$	4.25
5	-2.0	15.60	$[2\ 10]'$	4.90
6	-2.0	18.20	$[2\ 10]'$	5.55
7	-2.0	20.80	$[2\ 10]'$	6.20
8	-2.0	23.40	$[2\ 10]'$	6.85
9	-2.0	26.00	$[2\ 10]'$	7.50
10	-2.0	28.60	$[2\ 10]'$	8.15
11	-2.0	31.20	$[2\ 10]'$	8.80
12	-2.0	33.80	$[2\ 10]'$	9.45
13	-2.0	36.40	$[2\ 10]'$	10.10

The surrogate search results are presented graphically in Figures 7.4, 7.5, and 7.6. The results indicate that cargo up-load rapidly increases for the first few steps along the steepest ascent path, and begins to slow down around step 5 or 6. Throughput, on the other hand, increases initially, with the increase in arrival rate, but then levels off at step 3 and 4. The probability that an arriving aircraft diverts exhibits a steadily increasing value as the analytical model is evaluated along the steepest ascent path.

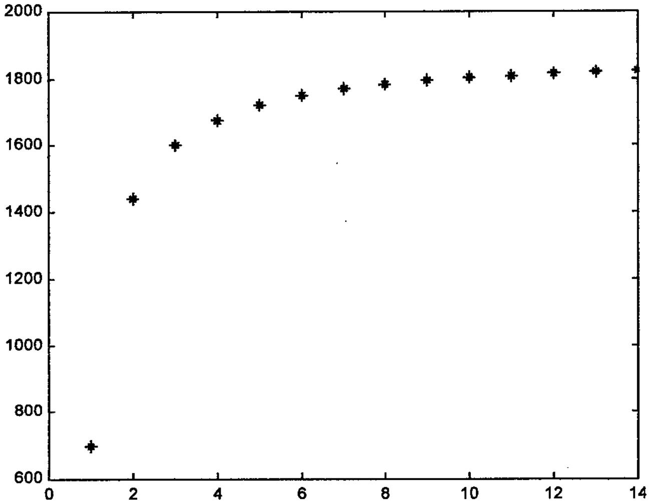

Figure 7.4 Surrogate search for cargo up-load (tons/24 hours).

As part of the study, AMC planners do not want the probability of diverting to exceed 0.05. From Figure 7.5 it appears that this threshold is crossed somewhere between steps 1 and 2. Here the resolution of the aircraft proportion within the analytical model did not allow for step sizes that could explore that region. So the aircraft arrival rate that results in a divert probability of 0.05 is not discernable from the surrogate search. At this point, a modified surrogate search is performed. Since the analytical model can be evaluated rapidly, performing more than one surrogate search is still cost efficient. Note that the time to complete one search at 20 different arrival rates is approximately 15 seconds on a 266 MHZ Pentium II PC versus approximately 10 minutes to perform 20 replications of Psuedo-BRACE at one design point.

Three new searches are conducted to locate the appropriate arrival rate by holding aircraft proportions constant while the arrival rate is varied. For each of the three searches, the aircraft proportion is held constant at $x_1 = 1/3, /1/4$, and $1/6$ respectively. The aircraft arrival rate is

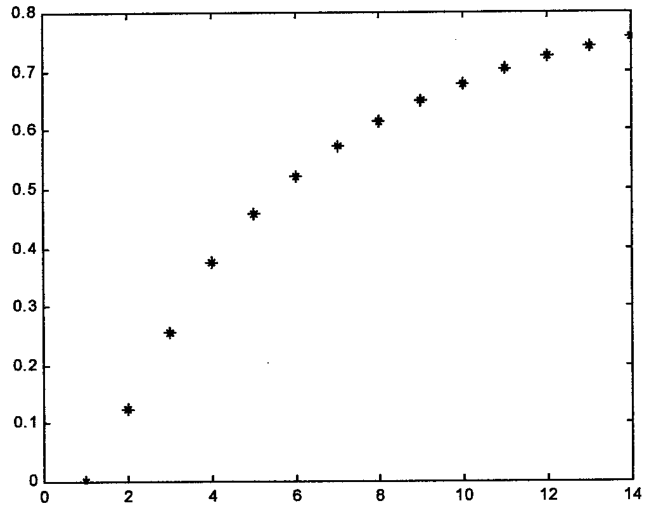

Figure 7.5 Surrogate search for P(Divert).

then varied in increments of 0.05 aircraft per hour starting at 1.30 aircraft per hour. The results of the searches for the probability of diverting and cargo up-load are presented graphically in Figures 7.7 and 7.8 respectively. The search results indicate that the 0.05 threshold for divert probability is crossed when the aircraft arrival rate is close 1.7 to 1.8 aircraft per hour depending on the aircraft mix. The results also indicate that for a given arrival rate, the maximum cargo up-loaded is provided when more C-B aircraft are used. Hence the speed of the analytical model is exploited to perform several surrogate searches in order to make up for a lack of resolution in the aircraft proportion treatment variable.

We now must validate the surrogate search results using Psuedo-BRACE. We will focus on the probability of diverting, since the problem statement constrains the system to operate with a probability of diverting to be less than 0.05. We choose to examine the results of the surrogate search when the proportion of C-A aircraft is 1/4. This translates to an analytical model setting of three C-A aircraft and 9 C-B aircraft. When the aircraft arrival rate is set to 1.75 aircraft per hour, the analytical model estimates $P(\text{Divert}) = 0.0502$. Validation of this result is examined by generating

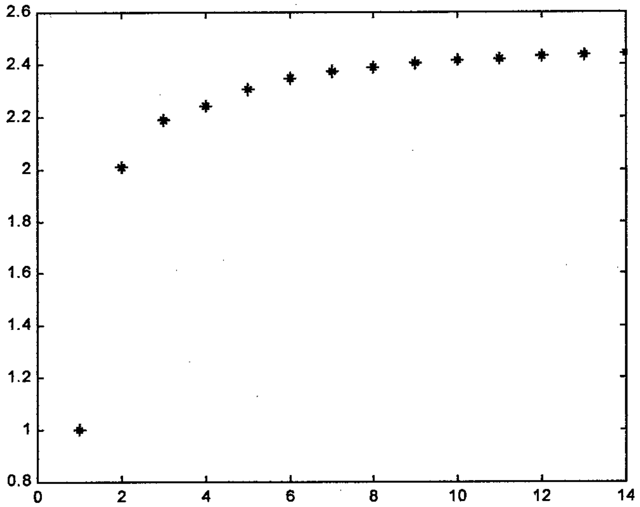

Figure 7.6 Surrogate search for throughtput (aircraft/hour).

20 replications of Pseudo-BRACE at the same design point. The estimated values for each of the three responses are compared to the results of the surrogate search at this design point in Table 7.13. The surrogate model produces responses for the mean cargo up-load and throughput that are reasonably close to the simulation model. However, the results for P(Divert) are unsatisfactory for prediction purposes. This is not totally unexpected since each model computes diverts differently. We expect more aircraft to divert in the analytical model since it assumes all aircraft arriving to a saturated airfield will immediately divert while the simulation model allows for a loiter time of 2 hours.

Table 7.13 Surrogate search verification for proportion of C-A aircraft = 1/4 and aircraft arrival rate = 1.75.

Response	$\bar{Y}_P^S$	90% C.I.	Z
Validation Point 1.			
Cargo	1259.8	[1247.6 1272.0]	1220.7
P(Divert)	0.011	[0.009 0.014]	0.050
Throughput	1.70	[1.68 1.71]	1.66

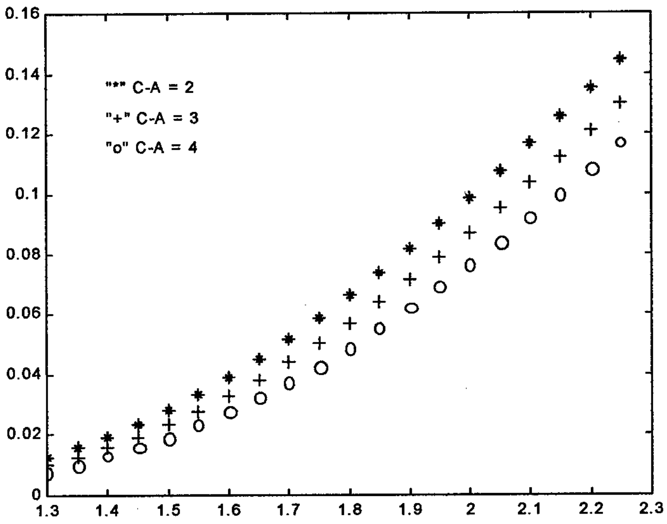

Figure 7.7 Secondary surrogate searches for P(Divert).

At this point, the analyst has essentially three options. The first is to use the current results to find an appropriate design point for further testing using the simulation model. A second option is to return to the conceptual analytical model development phase and modify the model so that it more closely approximates Psuedo-BRACE. The third, and recommended option is to adjust the output of the analytical model based on the results of the simulation replications at the tested design point. In a sense, we are creating a "new" analytical model using the results of the current model.

To adjust the analytical model output for P(Divert) to more closely approximate the simulation output, we first assume that the shape of the response surface is approximately the same for both models. Thus the difference recorded at the first validation point represents a constant difference, or translation, between the two surfaces. Using these assumptions, the first step is to approximate the response surface, or curve of the analytical model for the probability of diverting. We use the surrogate search results and the simulation response at validation point one to approximate the P(Divert) response curve for proportion of C-A aircraft equals $1/4$. By visually

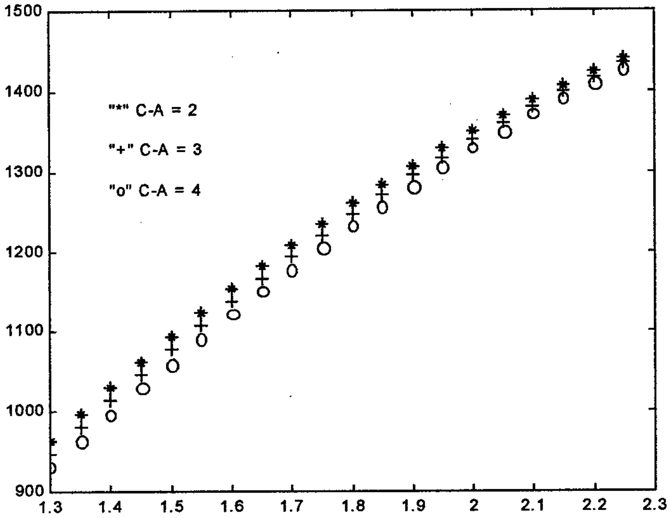

Figure 7.8 Secondary surrogate search for cargo (tons/24 hours).

inspecting the response curve, we decide to approximate it using a second order curve given by

$$P(\text{Divert})^{ss} = b_0^{ss} + b_1^{ss}\lambda + b_2^{ss}\lambda^2 \tag{7.14}$$

where λ is the aircraft arrival rate. Using least squares approximation results in

$$P(\text{Divert})^{ss} = 0.0584 - 0.1337\lambda + 0.0738\lambda^2 \tag{7.15}$$

To translate the approximated response curve, we set $\lambda = 1.75$ and $P(\text{Divert})^{ss} = 0.011$ based on the Pseudo-BRACE results. Holding b_1^{ss} and b_2^{ss} constant, we solve for a new b_0^{ss} constant. The final approximation is then

$$P(\text{Divert})^{ss} = 0.0190 - 0.1337\lambda + 0.0738\lambda^2 \tag{7.16}$$

Using the surrogate search based approximation, we solve the quadratic equation

$$0.05 = 0.0190 - 0.1337\lambda + 0.0738\lambda^2 \tag{7.17}$$

which has a solution of approximately 2.03 aircraft per hour. Based on this value, we generate 20 replications of Psuedo-BRACE with an aircraft arrival rate of 2.05 aircraft per hour. The results for each of the performance measures are in Table 7.14. The adjusted analytical model provides a much more accurate approximation of $P(\text{Divert})$. In fact, we have identified a point within the design space where the simulation model response where $P(\text{Divert}) = 0.050$ is contained within the 90 percent confidence interval. If desired, a new experimental design center could be placed at this design point for further study of the response surface.

Table 7.14 Surrogate search verification for proportion of C-A aircraft = 1/4 and aircraft arrival rate = 2.05.

Response	Y_P^S	90% C.I.	Z	Adjusted Z
Validation Point 2.				
Cargo	1440.6	[1425.0 1456.3]	1380.4	–
P(Divert)	0.046	[0.040 0.053]	0.104	0.055
Throughput	1.94	[1.92 1.96]	1.88	–

In summary, we have shown the effective application of the surrogate search method to a simple RSM study. The application demonstrates several characteristics of the surrogate search method. First, we show how the method is fully integrated within the context of an RSM study. Once a conceptual model is developed and translated to a computer program, the additional steps for operational validity are performed as each step of the study is performed. The flexibility of the surrogate search methodology has also been demonstrated. The simple study provided a situation where an inadequate initial surrogate search can be rapidly augmented with additional modified searches. In this case the modified searches were made with one of the treatments held constant. Finally, we demonstrate a simple means of adjusting the output of the analytical model when it is believed that the two models' response surfaces differ by only a constant. The method provides a rapid means of refining the surrogate search model output. In the next section, we demonstrate the surrogate search method on an actual Air Force simulation model of realistic size and complexity.

7.3 Airlift Flow Model RSM Study

We now present an application of the surrogate search method using the USAF Air Mobility Command (AMC) Airlift Flow Model (AFM) simulation model. We begin with a brief description of the airlift system and the AFM simulation model. Next, we discuss an AFM scenario that has been developed for academic research purposes. We then present an RSM study based on the academic scenario that will be examined using the surrogate search methodology. This is followed by a description of the particular settings of the AFM model in order to complete the study. We then present the analytical model used as a surrogate and the results of the surrogate search validation process. We conclude with the results of the study using the surrogate search method.

7.3.1 The Airlift System and AFM. AMC is responsible for providing global airlift of cargo and troops in support of the Department of Defense (DOD). The airlift system consists mainly of military aircraft, aircrews, airfields, air routes, air refueling, support equipment and personnel, fuel, and the airlift movement requirements. Under certain conditions, the National Command Authority can also task civilian airlines to provide aircraft in support of the airlift mission. These aircraft are referred to as the Civil Reserve Air Fleet (CRAF) and when activated fall under the authority of AMC. The airlift movement requirements for any particular DOD tasking, or plan, are contained within a Time-Phased Force Deployment Data (TPFDD) document. The TPFDD includes the on-load location, on-load availability day, required off-load location, and required delivery day for each requirement. It is then up to the AMC planners to plan the necessary airlift missions to meet the TPFDD requirements.

AMC uses the Mobility Analysis Support System (MASS) simulation model as an analysis tool to support decision making related to the airlift system. The core of the model is the Airlift Flow Model (AFM), which simulates the global airlift system. AFM is a stochastic discrete event simulation model. It is capable of simulating AMC policies, procedures, operations, aircraft, air bases, cargo, passengers, and support resources [11]. AFM simulates a fleet of aircraft moving a given amount of cargo and passengers from an on-load point, through any needed en-route stops, to an off-load point, then recovering and returning to home station for another mission. The model can continue this process for as many simulated days as desired, or until all requirements have been airlifted to their destination [11]. This process is referred to as executing a scenario. To accomplish a scenario, AFM performs three major tasks: simulation control, mission planning, and mission execution. The relationship between the three functions is depicted in Figure 7.9.

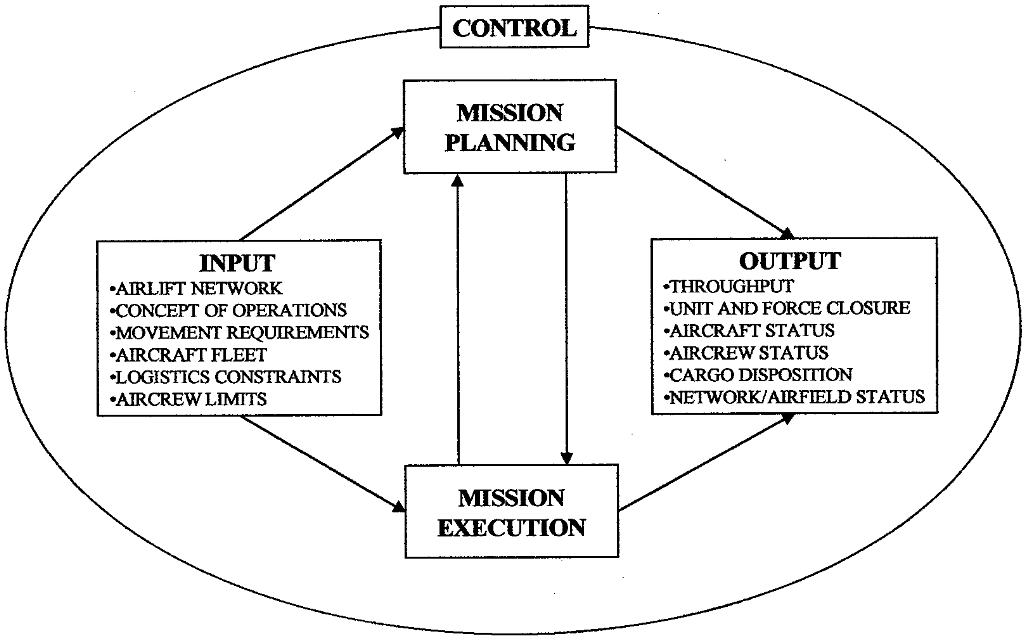

Figure 7.9 Airlift Flow Model (AFM) functionality relationship.

The following excerpt from an AMC Studies and Analysis Flight point paper [33] provides an excellent synopsis of AFM:

AFM Inputs Possible AFM inputs include:

- A TPFDD document containing airlift movement requirements.
- An airlift network consisting of on-loads, off-loads, en-route stops, recovery bases, and home stations connected by user-defined routes.
- An airlift fleet of different aircraft types identified by individual tail numbers.
- Individual aircrews who must be available to allow missions to be flown.
- Logistic factors which account for refueling, maintenance, and material handling of cargo.
- Concepts of operations that include strategic inter-theater airlift, aerial refueling, intra-theater shuttle operations, direct delivery operations, and recovery/stage operations.

Mission Planning AFM performs the following mission planning tasks:

- Prioritization of requirements by available-to-load dates and required delivery dates.
- Prioritized route selection and reservation for flight planning.
- Marrying a specific aircraft tail number to the next eligible requirement.
- Crew planning to ensure that only the crews eligible to fly do fly.

Mission Execution AFM simulates mission execution by simulating:

- Typical sortie events including: taxi-out, takeoff, departure, in-route cruise, initial approach, final approach, landing, taxi-in, and ground activities for every sortie of the mission.
- Ground activity resource allocation and planned delays for: ramp space, off-loading and on-loading cargo, refueling, maintenance, and crew changing.
- Optionally, detailed loading of each piece of cargo for compatibility with the doors and remaining space on each aircraft.

- Crew activities and monitoring events, including: crew rest, crew monthly and quarterly flying hour limits, crew availability, and searches for unavailable crews.

AFM Outputs AFM output includes:

- Aircraft related statistics such as: utilization rates, payload, ground service time, flight time, and system delays.
- Aircrew related statistics such as: crew duty days, number of crews, hours flown by each crew, and crew availability.
- Cargo related statistics: total tons delivered, tons per day throughput, unit and force closure, actual million tons miles per day flown, and cargo remaining in backlog.
- Airlift network statistics: typical cycle times, flying times, network airfield use, maximum on ground (MOG) constraints, and system bottlenecks.

7.3.2 AFM Academic Scenario. We begin by examining a notional AFM strategic intertheater airlift scenario. Since almost all scenarios studied by AMC contain classified information, an academic scenario of realistic size and complexity has been developed. The scenario's nominal aircraft fleet consists of 185 total aircraft, including C-5's, C-17's, C-141's, and CRAF aircraft. The TPFDD contains an airlift requirement of approximately 26,000 tons of cargo and 35,700 passengers, which become available for movement over a period of 15 days. The primary onload locations are McGuire and Charleston Air Force Bases (AFB's) and John F. Kennedy (JFK) International Airport. The primary off-load locations are in Bahrain, Dhahran, and King Abdul Aziz International Airports in Saudi Arabia. All requirements are to be delivered within 21 days. The airlift missions beginning in the CONUS are either flown directly to their destination or are routed through one of the several en-route air bases located in Europe. The en-route air bases for the military aircraft are Mildenhall Air Base (AB), England; Ramstein AB, Germany; Moron AB, Spain; and Rota Naval Air Station, Spain. The CRAF aircraft use Heathrow International Airport, England and Frankfurt Main International Airport, Germany as their en-route bases.

The nominal aircraft fleet consists of 60 C-5's, 50 C-17's, 50 C-141's, and 25 CRAF aircraft. Half of these aircraft become available for missions on the first day of the scenario with the other half available on the second day. The home base for half of the C-141's and C-5's is McGuire AFB with the remaining aircraft's home base is Charleston AFB. All of the C-17's are based at Charleston AFB and all of the CRAF aircraft have JFK as their home station.

In the real airlift system, the availability of different resources at each base determines the rate at which aircraft are serviced. These resources include parking spaces, fuel, parts, support equipment, and different support personnel and AFM has the ability to simulate these different resources. For this scenario these resources, other than fuel, are aggregated within the single resource called *maximum on the ground* (MOG). For each air base in the scenario, a single value of MOG is input representing the maximum number of aircraft that an air base can support at one time. Any aircraft that arrives at a base such that the number of aircraft on the base now exceeds the MOG value for that base must wait for a MOG resource to be freed before any servicing is begun on that aircraft. All servicing is then based on a constant time for the type of servicing being performed (off-load, en-route, on-load, etc.) and the type of aircraft. Based on this, the mission planning function of AFM must reserve MOG at the air bases necessary to complete a particular mission before a mission can be planned [11]. In this scenario, only McGuire AFB, Charleston AFB, and the civilian airports have unlimited MOG.

For every hour an aircraft flies, a certain amount of ground servicing and repair time is required before the aircraft is capable of flying again. The long run average of time a particular type of aircraft is capable of flying in a 24 hour period of time is called the aircraft's *capability rate.* Several factors determine this capability rate for each type of aircraft. The factors include aircraft system reliability, scheduled maintenance requirements, spare part availability, and average ground servicing times. Since AFM uses a constant ground time for ground servicing in this scenario, some other means of simulating this capability rate is required. These aircraft capability rates are simulated in AFM in the academic scenario using two procedures, the capping procedure and the differential procedure.

The AFM capping procedure relies upon a user input of aircraft capability rates based on historical data. Based on the number of aircraft in the scenario, the capping procedure determines the number of available flying hours for each day. It then monitors the actual flying hours flown, and planned, during AFM execution. Once the flying hour cap is reached, the capping procedure prevents the planning of any more missions until more flying hours become available. This procedure can work well, however there are some situations where the application of the procedure results in unrealistic aircraft activity [11].

The differential procedure was developed to address the problems encountered in the capping procedure [11]. This procedure randomly grounds aircraft at their home station so that the amount of flying accomplished tends toward the input capability rate. The differential procedure accomplishes this by tracking the history of all home station visits realized during an AFM replication. Based on the observed history, aircraft are periodically grounded at their home station based on the solution to a transportation control problem that determines the amount of additional ground time required to meet the desired capability rate [11]. Both procedures are then applied to realistically simulate the historical aircraft capability rates.

7.3.3 RSM Problem Statement. Of the many possible performance measures for this scenario, we focus on the delivery of cargo and passengers. We can measure cargo and passengers with a single value by considering passengers and their gear to weigh 350 pounds each. Therefore, for the rest of this section we consider all requirements airlifted as tons of cargo. An obvious problem statement that we first considered is that of finding the optimal mix of aircraft that maximizes the tons of cargo delivered per day. However, there are some inherent problems with that type of problem statement in the context of this scenario.

For the academic scenario, and any real-world scenario of this type, there are two factors, other than the amount of aircraft, which limit the amount of cargo that can be delivered each day. The first factor is the TPFDD requirement. Obviously, only a finite amount of cargo is available for movement on any given day as defined by the TPFDD document. The second factor is the amount of MOG available throughout the airlift network. In other words, there is a finite limit to the number of aircraft that can be serviced and/or off-loaded at any given time at the en-route and delivery destination air bases. Therefore, as the number of aircraft in the scenario is increased, one of these two factors will act as a limit to the amount of cargo that can be moved per day. If the network can handle it, that limit will be defined by the TPFDD. Otherwise, the constrained capacity of the airlift network will limit the amount of cargo delivered per day at a level less than that made available by the TPFDD.

While the maximization of cargo delivery rates is important, it is not necessarily the goal that AMC strives to meet during a particular airlift tasking. Actually, the primary goal is to deliver the cargo where it is required, when it is required, as defined by the TPFDD. In order to meet the academic scenario's delivery requirements, it is not necessary to deliver the cargo at the same rate that it is made available. This is clear since all cargo is available for movement in 15 days while it's delivery requirement occurs typically 5 days later. Therefore, we can redefine the limiting factors in terms of meeting the TPFDD delivery requirements. During our initial investigations of the academic scenario we discovered that the network can service enough aircraft so that the TPFDD delivery requirement can be met. Further, we discovered that the nominal aircraft fleet of 185 aircraft contains excess capacity. By that we mean the TPFDD delivery requirement can be met with less than 185 aircraft.

Based on the above discussion, we wish to find the optimal mix of aircraft that most "efficiently" delivers the TPFDD requirement on time. The task then is to define an appropriate measure of effectiveness that rewards on-time or early delivery of cargo and penalizes excess airlift capacity. Hence the measure should be a function of the number of aircraft, the amount of cargo delivered, the amount of time that it takes to deliver the cargo, and the TPFDD required delivery time. Based on this we develop two measures of effectiveness described below.

If we wish to maximize our efficient delivery of cargo, we can translate that to a desire to maximize the amount of on-time, or early, cargo each aircraft delivers. To do this we measure the average difference between the accumulated delivered cargo and the required accumulated cargo per day, per aircraft. We call this the *early cargo per aircraft* (ECA) measure of effectiveness. If we are at, or near, the maximum throughput of the network/TPFDD constraints, adding more aircraft will decrease the value of ECA. Conversely removing aircraft will increase the value of ECA until the cargo throughput drops below some threshold. Thus, ECA provides a measure of the efficient effectiveness of the aircraft fleet.

Another approach is to multiply the number of aircraft in the fleet by a measure of their effective efficiency. Thus, our goal is to minimize this measure, which we call the *aircraft-cargo ratio* (ACR). To measure the effective efficiency we consider two ratios. The first ratio compares the amount of time realized to close out a TPFDD versus the required close out time. If the TPFDD is closed out in less time than required, this ratio is less than one, reducing the value of the ACR. Taking longer than planned causes the ratio to be greater than one and ACR grows. The second ratio compares the amount of cargo in the TPFDD to the amount that is actually delivered by the last required delivery date. If all the cargo is moved by the last day of the TPFFD, this ratio will equal one. Otherwise, if some of the cargo is delivered after the last required delivery date, this ratio will be greater than one causing an increase in the ACR. We must make an adjustment to this measure to account for situations where the total number of aircraft in the fleet is particularly low, causing a "false" low value of ACR. When the amount of time required to close out the TPFDD exceeds the required time we add additional "aircraft" to the value of ACR. The number of additional aircraft added is equal to the amount of cargo undelivered by the TPFDD close out date, divided by the average aircraft cargo load observed. In this manner, the effective efficiency of the aircraft fleet is measured by the ACR.

We now derive AFM simulation output statistics to estimate ECA and ACR. The method of independent replications is used to estimate each of the measures and the notation is developed with that method in mind. We let the number of replications generated at a design point equal n. Since the two measures share many of the same terms, we develop the following shared notation. First let the number of aircraft in the aircraft fleet used in a scenario, or experimental design point, be denoted by A. We now address the TPFDD cargo requirements, which are constant throughout our study. Each line of the TPFDD denotes a consolidated "package" of cargo and/or passengers available at a single airbase that is required at a single airbase in the theater of operations. The package is assigned a date (number of days from the beginning of the scenario) when it is available for loading on an aircraft, and a date when it is required for delivery. We track the amount of cargo that is required for delivery each day by first letting i represent the scenario day such that $1 \leq i \leq 21$. Then let $c_R(i)$ be the amount of cargo, in tons, that the TPFDD designates for delivery on day i over the entire airlift system. Thus the accumulated TPFDD cargo requirement for the k^{th} day of the scenario, $C_R(k)$, is given by

$$C_R(k) = \sum_{i \leq k} c_R(i), \quad k = 1, 2, \ldots, 21 \tag{7.18}$$

In a similar manner, we represent the cargo that is actually delivered on day i during replication j as $c_D^S(i)_j$. Thus, the accumulated cargo delivered by day k in the scenario for replication j, $C_D^S(k)_j$, is computed as

$$C_D^S(k)_j = \sum_{i \le k} c_D^S(i)_j, \quad k = 1, 2, \ldots, 21 \;\; j = 1, 2, \ldots, n \tag{7.19}$$

We now represent the number of days required to deliver the entire TPFDD cargo requirement for replication j as T_{Dj}^S, which is defined mathematically as

$$T_{Dj}^S = k \;\ni\; \left\{k = 1, 2, \ldots \mid C_D^S(k)_j = C_R(21) \text{ and } C_D^S(k-1)_j < C_R(21)\right\} \tag{7.20}$$

We also let τ_{Dj}^S represent the number of hours required to deliver all cargo for replication j such that $24T_{Dj}^S - 24 < \tau_{Dj}^S \le 24T_{Dj}^S$.

We estimate early cargo per aircraft, ECA, using AFM in the following manner. For each replication j we calculate the realized ECA_j^S by

$$ECA_j^S = \frac{1}{21A} \sum_{i=1}^{T_{Dj}^S} \left\{C_D^S(i)_j - C_R(i)\right\}, \quad j = 1, 2, \ldots, n \tag{7.21}$$

Thus the estimate of ECA using the method of independent replications is given by

$$\overline{ECA}^S = n^{-1} \sum_{j=1}^{n} ECA_j^S \tag{7.22}$$

The aircraft-cargo ratio, ACR is estimated by first considering the effective efficiency ratios. First consider the ratio of the time required to deliver the TPFDD requirement during replication j to the amount of time before the TPFDD requires all cargo to be delivered, D_j^S, found by

$$D_j^S = \frac{\tau_{Dj}^S}{24 \cdot 21}, \quad j = 1, 2, \ldots, n \tag{7.23}$$

Now consider the ratio of the total cargo requirement to the accumulated cargo delivered during replication j given by

$$M_j^S = \frac{C_R(21)}{C_D^S(21)_j}, \quad j = 1, 2, \ldots, n \tag{7.24}$$

If we let G_j^S be the average cargo load of all cargo deliveries in replication j, the estimated ACR for replication j is calculated as

$$ACR_j^S = \begin{cases} A(D_j^S)(M_j^S) & \text{if } T_{Dj}^S \leq 21 \\ A(D_j^S)(M_j^S) + \frac{C_R(21) - C_D^S(21)_j}{G_j^S} & \text{if } T_{Dj}^S > 21 \end{cases} \tag{7.25}$$

so that ACR can be estimated by

$$\overline{ACR}^S = n^{-1} \sum_{j=1}^{n} ACR_j^S \tag{7.26}$$

We perform an RSM study using both measures of effectiveness, where we wish to maximize ECA and minimize ACR. We don't know if the optimal value for each of these two measures occurs at the same design point, so our goal is to construct a response surface that contains the optimal value for each measure. The design factors, or treatments, for the study are the number of each of the four types of aircraft in the aircraft fleet. We assign the factors by

$$\begin{aligned} x_1 &= \text{number of CRAF aircraft} \\ x_2 &= \text{number of C} - 141 \text{ aircraft} \\ x_3 &= \text{number of C} - 5 \text{ aircraft} \\ x_4 &= \text{number of C} - 17 \text{ aircraft} \end{aligned} \tag{7.27}$$

To construct an experimental design to estimate the first order response surface for each measure we construct a 2^4 factorial design in the following manner. Using the academic scenario nominal settings, we assign the design center as $x_1 = 25$, $x_2 = 50$, $x_3 = 60$, and $x_4 = 50$, where we vary x_1 by plus and minus 5 aircraft, and the other three treatments by plus and minus 10 aircraft each. The levels of the coded treatment variables, $\Theta = (\Theta_1, \Theta_2, \Theta_3, \Theta_4)'$ are found using the following formulas

$$\Theta_1^j = \frac{x_1^j - 25}{5}, \quad j = H, L \tag{7.28}$$

$$\Theta_2^j = \frac{x_2^j - 50}{10}, \quad j = H, L \tag{7.29}$$

$$\Theta_3^j = \frac{x_3^j - 60}{10}, \quad j = H, L \tag{7.30}$$

$$\Theta_4^j = \frac{x_4^j - 50}{10}, \quad j = H, L \tag{7.31}$$

The resulting coded and uncoded experimental design appears in Table 7.15.

Table 7.15 Initial AFM RSM 2^4 factorial design.

	Coded				Uncoded			
Design	Treatment Levels				Treatment Levels			
Point	Θ_1	Θ_2	Θ_3	Θ_4	x_1	x_2	x_3	x_4
1	1	1	1	1	30	60	70	60
2	1	1	1	-1	30	60	70	40
3	1	1	-1	1	30	60	50	60
4	1	1	-1	-1	30	60	50	40
5	1	-1	1	1	30	40	70	60
6	1	-1	1	-1	30	40	70	40
7	1	-1	-1	1	30	40	50	60
8	1	-1	-1	-1	30	40	50	40
9	-1	1	1	1	20	60	70	60
10	-1	1	1	-1	20	60	70	40
11	-1	1	-1	1	20	60	50	60
12	-1	1	-1	-1	20	60	50	40
13	-1	-1	1	1	20	40	70	60
14	-1	-1	1	-1	20	40	70	40
15	-1	-1	-1	1	20	40	50	60
16	-1	-1	-1	-1	20	40	50	40

7.3.4 AFM Settings. We describe the specific aircraft and airfield parameters addressed above in section 7.3.2. We begin by listing the aircraft parameters in Table 7.16. The parameters include type of body (wide or narrow), capability rate, and the standard times associated with off-load servicing, on-load servicing, and en-route servicing. The body type determines the amount of ramp space and associated MOG that an aircraft occupies. Essentially, a wide body aircraft occupies twice as much MOG as a narrow body aircraft. The capability rate indicates the long range average operating time for every 24 hours. Recall that the AFM capping and differential procedures use the capability rate input to simulate aircraft operations. For the standard service times, the en-route time is also used as the standard recovery time at an aircraft's recovery or home base. The MOG capacity for each air base in the scenario, by aircraft body type, is listed in Table 7.17. Each of these settings remain constant at each of the 16 experimental design points.

Table 7.16 AFM RSM study aircraft parameters.

Aircraft	Body	Capability	Standard Service Times (hrs)		
Type	Type	Rate (hrs)	En-Route	Off-Load	On-Load
CRAF	Wide	12.0	1.50	2.00	3.50
C-141	Narrow	12.2	2.25	2.25	2.25
C-5	Wide	10.7	3.25	3.25	4.25
C-17	Narrow	15.3	2.25	2.25	2.25

Table 7.17 AFM RSM study airbase MOG capacities.

	MOG	
Air Base	Narrow	Wide
Charleston AFB	Unlimited	Unlimited
McGuire AFB	Unlimited	Unlimited
JFK IAP	Unlimited	Unlimited
Ramstein AB	9	4
Mildenhall AB	4	2
Barajas IAP	28	14
Moron AB	4	4
Rota NAS	2	1
Frankfurt Main IAP	Unlimited	Unlimited
Heathrow IAP	32	16
Bahrain IAP	26	13
Dhahran IAP	10	5
King Abdul Aziz IAP	13	6

7.3.5 Surrogate Search Validation and Initial RSM Results. The surrogate search validation and initial RSM results are presented in the following section. We begin by describing and validating the proposed conceptual analytical model of the AFM simulation model. Next we establish surrogate search operational validity while performing and analyzing the initial experiments of the RSM study.

7.3.5.1 AFM Conceptual Analytical Model Validity. The next step in the surrogate search methodology is to propose and validate a conceptual analytical model of the credible simulation model AFM. As presented in Section 6.3.1 we validate the conceptual analytical model by first establishing face validity and then ensure that the analytical model possesses the appropriate output performance measures and treatment parameter inputs. See Figure 7.1 for a flow chart of the described process. We begin by first describing the proposed conceptual model of AFM. We then examine the choices made in creating the model as we attempt to establish face validity of the model. This section is then concluded with an investigation into the presence of the appropriate output and input parameters.

We propose a multi-class closed queueing network model solved via the MVA algorithm as the conceptual analytical model of AFM. The customers, or entities, within the network are the airlift aircraft simulated in AFM. We initially consider four classes of customers corresponding to the four types of aircraft in the AFM scenario. Thus, for the RSM design center we have a total of 185 customers spread appropriately across the four classes. We begin by representing each of the 13 air bases as a single service center with an infinite number of servers. The service time at each of the air bases is established by observing the realized mean time aircraft spend at the appropriate air base during AFM realizations. The air routes from one air base to another are also represented as a service center with an infinite number of servers (or equivalently as a delay station [8]) where the mean service time is found by observing realizations of the AFM simulation model. For the 13 air bases there are possibly $(13-1)^{13} = 12^{13} = 156$ possible air routes for each class of aircraft resulting in a total of $12^{13} + 13 = 169$ possible service centers for each aircraft class. Based upon an input to AFM of permissible aircraft routings, and initial observations of AFM, we don't expect the conceptual model to require all 169 service centers, however the model size will be fairly large none the less. As with the mean service times, the probabilities that define the four probability transition matrices of this model are determined by observing AFM simulation realizations.

A primary concern when creating an analytical model for the purposes of surrogate searches and generating an ACV is the relative speed at which the ACV is computed when compared to the simulation model. Initial testing of this analytical model at design point one with 215 aircraft (customers) in 4 classes resulted in computation times of approximately 194 minutes. Since 10 replications of the AFM model at a design point only takes approximately 15 minutes to generate, the computation time of the 4 class analytical model is unsatisfactory. It is well known that the number of numerical operations when computing the MVA algorithm grows exponentially with the number of customer classes [16]. Therefore, we choose to reduce the number of customer classes from 4 to 2 by grouping all of the military aircraft into a single class. This is a natural choice since the military aircraft share the same home stations and en-route bases and the CRAF aircraft have a different home station and set of en-route bases. Making this adjustment reduces computation time at the same design point from 194 minutes to 8.6 seconds.

We now consider whether the proposed conceptual analytical model is a reasonable model of AFM and thus meets the requirement of face validity. Given that we can correctly identify the model parameters, the basic assumption that AFM can be modeled as a closed queueing network certainly seems reasonable. After all, in AFM we have aircraft (customers) flowing through a closed airlift system consisting of 13 air bases and the appropriate set of air routes that connect them. However, the performance measures calculated by the MVA algorithm are based on the assumption that the closed queueing network has reached "steady state" operation [29]. The output statistics of the AFM model are the result of a terminating simulation model that doesn't necessarily begin to exhibit steady state behavior. Indeed, the output statistics include data from the beginning of each replication when AFM is certainly not operating at steady state. The alternative is to construct an analytical model that doesn't assume steady state behavior, however finding a solution for non-steady state queues is difficult for even the simplest of queueing models [28]. Even if such a solution could be found for AFM, it is assumed that the complexity of such of model would prohibit it's efficient use. Thus, constructing an analytical model that approximates AFM using the MVA algorithm approach seems to be the best alternative.

We now investigate the reasonableness of the settings and parameters of the proposed closed queueing network. It certainly seems appropriate that the air routes are modeled as delay stations with service times equal to the realized transit times of the AFM model. What is not obvious is whether or not the air bases are appropriately modeled as delay stations and if the decision processes involved in creating airlift missions can be adequately modeled by a transition probability matrix based on the observed routings in AFM. Further, even if these modeling decisions are appropriate, we must also decide if the grouping of all military aircraft into a single class is appropriate.

When an aircraft arrives at an airbase that has an available MOG resource to either on-load or off-load cargo, or for en-route servicing, it spends a fixed amount of time at that air field based on the standard servicing times in Table 7.16. As long as there is an available MOG resource, the aircraft doesn't wait (queue) for service and the time in service is the same regardless of the number of aircraft also in service. Further, if the modeler carefully selects the AFM input parameters for allowable aircraft routings, AFM will schedule missions and reserve MOG so that each air base will have enough MOG available for all arriving aircraft. Therefore, it certainly seems appropriate to model the en-route and off-load bases as delay stations.

Aircraft behave differently at their home stations. Two factors interact to determine the amount of time an aircraft spends at home station. The first factor is the combined capping and differential procedure that ground aircraft at their home station in order to simulate the aircraft capability rate. The other factor consists of waiting in a first-in-first-out queue of all aircraft at each of the three home stations for an airlift mission to be planned. The planning of a mission depends on the availability of cargo and the availability of sufficient MOG to complete the mission [11]. In this case it not clear if modeling the home stations as delay stations is appropriate. However, it is also not clear how to model it differently. The MVA algorithm cannot handle a queue of customers from more than one service station [29]. So at this point, we decide to use the delay station model at all air bases and test our decision during surrogate search operational validation.

The routing of aircraft within AFM is a complex process. As mentioned above, it is a function of available aircraft flying hours, availability of cargo, the location of the cargo, and the availability of MOG in the airlift network, among other things. In addition, a planned airlift mission includes all the bases that will be visited. Conversely, the Markovian transition probability matrices used by the conceptual analytical model are constructed on the assumption that the probability that a customer (aircraft) moves from one station to another doesn't depend on its prior location. However, given that the TPFDD requirement remains constant for all replications, a constant proportion of the cargo at each starting location must be airlifted to each of the destination locations it is not unreasonable to expect some amount of regularity for the routing of airlift missions from replication to another. Hence these realized routing proportions could logically serve as approximations of the airlift planning process. Hence, we consider the conceptual analytical model as face valid and proceed to the next aspect of conceptual analytical model validation: input and output matching.

We will first attempt to create output performance measures from the analytical model for estimating ECA and ACR. Note that both of these performance measures are functions of the number of aircraft assigned, the cargo delivered every day, and the TPFFD requirements. For the analytical model, we observe that the average amount of cargo delivered per day by each type of aircraft is a function of the average throughput of aircraft leaving the CONUS air bases and the average aircraft cargo load.

We define the following terms in order to compute the average cargo delivered per day. Let $B = \{1, 2, 3\}$ be the set of CONUS home stations such that JFK IAP is base 1, Charleston AFB is base 2, and McGuire AFB is base 3. Also, let $Y = \{C, M\}$ be the set of aircraft types in the scenario so that C represents CRAF aircraft and M is military aircraft. Then $\mathbf{N}^A = (N_C^A, N_M^A)'$ represents the number of CRAF and military aircraft in the scenario respectively where

$$N_M^A = N_{\text{C}-141} + N_{\text{C}-5} + N_{\text{C}-17} \tag{7.32}$$

is the sum of the three types of military aircraft used in the corresponding AFM scenario. The average throughput per hour for base $i \in B$ as calculated by the analytic model is given by $\boldsymbol{\lambda}_i = (\lambda_i^C, \lambda_i^M)'$ for CRAF and military aircraft respectively. Since aircraft leaving their home station will only fly to another home station if they are going to pick-up cargo, we consider only those aircraft flying to bases other than the other home station as carrying cargo. Thus, to calculate cargo throughput, let $p_{i,j}^C$ represent the probability that a CRAF aircraft leaving base $i \in B$ will fly to base $j \in B$, $j \neq i$ and $p_{i,j}^M$ represents the same probability for military aircraft. Thus the probability that an aircraft of type $y \in Y$ leaves base i to deliver cargo, $P_D^y(i)$, is given by

$$P_D^y(i) = 1 - \sum_{\substack{j \in B \\ j \neq i}} p_{i,j}^y, \qquad y \in Y,\ i \in B \tag{7.33}$$

Then the average throughput per hour for aircraft of type $y \in Y$ from station $i \in B$ delivering cargo, Λ_i^y, is found by

$$\Lambda_i^y = \lambda_i^y \left\{ P_D^y(i) \right\}, \qquad y \in Y,\ i \in B \tag{7.34}$$

If we let G_{CRAF}, $G_{\text{C}-141}$, $G_{\text{C}-5}$, and $G_{\text{C}-17}$ be the average cargo load in tons for each type of aircraft as realized by AFM, we can compute the average tons of cargo delivered every hour by aircraft type C for station $i \in B$, γ_i^y, by

$$\gamma_i^C = \Lambda_i^C \left(G_{\mathrm{CRAF}}\right), \quad i \in B \tag{7.35}$$

and for aircraft type M by conditioning on the each type of military aircraft given by

$$\gamma_i^M = \Lambda_i^C \left(N_M^A\right)^{-1} \left[N_{\mathrm{C-141}} \; N_{\mathrm{C-5}} \; N_{\mathrm{C-141}}\right] \left[G_{\mathrm{C-141}} \; G_{\mathrm{C-5}} \; G_{\mathrm{C-17}}\right]', \quad i \in B \tag{7.36}$$

Thus, the average tons of cargo delivered per day as computed by the analytical model, Γ^A, is given by

$$\Gamma^A = 24 \sum_{y \in Y} \sum_{i \in B} \gamma_i^y \tag{7.37}$$

Given the average tons of cargo delivered per day, we now define the formulas for computing the analytical performance measures ECA^A and ACR^A. To compute these performance measures, we must determine the amount of cargo that is delivered each day of a scenario. In AFM, this is a straightforward process. We need only record the values realized during each realization of the simulation model. For the conceptual analytical model, it is not a straightforward process. First of all, the procedure outlined above provides us with an expected value for the amount of cargo delivered each day, Γ^A, under steady state assumptions and doesn't provide us with the probability distribution of that value. Further, the amount of cargo "moved" by the analytical model is not constrained by a TPFDD. Therefore, it is possible to compute a value of Γ^A that exceeds the amount of cargo available for movement by the TPFDD on any given day. Despite these inherent problems, we propose a method for estimating ECA^A and ACR^A using the analytical model.

We make two comments about our derivation of ECA^A and ACR^A before we begin. The first comment is that these formulas were developed in an iterative process by comparing analytical model results for several proposed formulas and methods to the results of AFM at three test design points. In order to save space and time we only present our final results. Our second comment is that there are several possible approaches that an analyst could use to approximate these performance measures. We are in a difficult situation in that we are attempting to use an analytical model that computes mean performance measures based on assumptions of steady state operation to approximate the behavior of a terminating simulation model. We have developed an approach that we find provides satisfactory results for our purposes.

In determining ECA^A and ACR^A we adjust the TPFDD requirements in order to mimic the mean performance measures made available by the analytical model. We first assume that the amount of cargo made available for airlift on the k^{th} day, $V^A(k)$ is divided equally across the 16 days defined in the TPFDD and the required delivery amounts, $c_D^A(k)$ are also divided equally between the 5^{th} and 21^{st} day of the plan, with the following exception. It became apparent that the amount of cargo delivered in the first two days during an AFM realization is always much less than that observed over the remainder of the replication. Based on this observation, we use the observed values of the AFM replications to set the availability of cargo for the first two days. Thus, for n replications, we set the mean cargo available in the analytical model for days 1 and 2 by

$$V^A(1) = n^{-1}\sum_{j=1}^{n} c_D^S(1)_j \quad (7.38)$$

$$V^A(2) = n^{-1}\sum_{j=1}^{n} c_D^S(2)_j \quad (7.39)$$

so that the cargo available for movement in the analytical model is given by

$$V^A(k) = \frac{C_R(21) - (V^A(1) + V^A(2))}{14}, \quad k = 3, 4, \ldots, 16 \quad (7.40)$$

In a similar manner, the amount of cargo that is required for delivery each day for the analytical model, $c_R^A(k)$ is found by

$$c_R^A(k) = \begin{cases} 0, & k = 1, 2, \ldots, 5 \\ \dfrac{C_R(21)}{16}, & k = 6, 7, \ldots, 21 \end{cases} \tag{7.41}$$

and the accumulated cargo delivery requirements are

$$C_R^A(k) = \sum_{i \le k} c_R^A(i), \quad k = 1, 2, \ldots, 21 \tag{7.42}$$

To determine the amount of cargo delivered each day by the analytical model we begin by making the assumption that the amount of cargo delivered on day $k = 1, 2, \ldots, 21$ by the analytical model, $c_D^A(k)$ is bounded above by Γ^A and bounded below by the amount of cargo available for delivery. In AFM, there is obviously a delay between the time that cargo is loaded onto an aircraft until it is delivered. The result of this delay is that some portion of the cargo delivered on any given day is on-loaded that same day while the remainder is on-loaded the previous day. Based on this observation, we limit the cargo delivered on the same day it is on-loaded to $\Gamma^A/2$. Thus, the cargo delivered on the first day of the scenario, $c_D^A(1)$ is given by

$$c_D^A(1) = \min\left\{\Gamma^A/2,\ V^A(1)\right\} \tag{7.43}$$

We track any cargo from the first day that is not delivered, or the *backlog* cargo, $b(1)$, by

$$b(1) = V^A(1) - c_D^A(1) \tag{7.44}$$

Then for the rest of the scenario, we determine the cargo delivered each day in the following manner. First we find the amount of cargo that is on-loaded the previous day for delivery on the current day represented by $\varphi(k)$. This is equal to the backlog cargo from the previous day, up to the mean cargo throughput Γ^A. Mathematically, this is formulated as

$$\varphi(k) = \min\left\{b(k-1),\, \Gamma^A\right\}, \quad k = 2, 3, \ldots \tag{7.45}$$

where

$$b(k) = b(k-1) - \varphi(k), \qquad k = 2, 3, \ldots \tag{7.46}$$

Next we determine $\rho(k)$, the remaining capacity to on-load and deliver cargo on the same day given by

$$\rho(k) = \min\left\{\Gamma^A - \varphi(k),\, \Gamma^A/2\right\}, \quad k = 2, 3, \ldots \tag{7.47}$$

which is limited by the amount of cargo that is actually made available for on-loading on that day so that the actual cargo on-loaded and delivered on the same day is given by

$$\delta(k) = \min\left\{\rho(k),\, V^A(k)\right\}, \quad k = 2, 3, \ldots \tag{7.48}$$

Thus we can now compute the cargo delivered on day k by

$$c_D^A(k) = \varphi(k) + \delta(k), \qquad k = 2, 3, \ldots \tag{7.49}$$

As before, the accumulated delivered cargo by each day is given by

$$C_D^A(k) = \sum_{i \le k} c_D^A(i), \qquad k = 1, 2, \ldots \tag{7.50}$$

This process continues until the analytical model TPFDD cargo requirement is completely delivered. The day that the TPFDD is closed out, T_D^A, is given by

$$T_D^A = k \ni \{k = 1, 2, \ldots \mid C_D^A(k) = C_R(21) \text{ and } C_D^A(k-1) < C_R(21)\} \tag{7.51}$$

The amount of time required, in hours, to deliver all cargo, τ_D^A is given by

$$\tau_D^A = 24\left(T_D^A - 1 + \frac{c_D^A(T_D^A)}{\Gamma^A}\right) \tag{7.52}$$

Given the values of $C_D^A(k)$ and T_D^A computed in the manner described above, we estimate ECA^A and ACR^A as follows. ECA^A is found by

$$ECA^A = \frac{1}{21A}\sum_{i=1}^{T_D^A}\{C_D^A(i) - C_R^A(i)\} \tag{7.53}$$

To compute ACR^A we first find the ratio between the amount of time required to close out the TPFDD to the amount of time the TPFDD requires all cargo to be delivered by, D^A which is defined as

$$D^A = \frac{\tau_D^A}{24 \cdot 21} \tag{7.54}$$

and the ratio of the total cargo requirement to the accumulated cargo delivered by the last required day given by

$$M^A = \frac{C_R(21)}{C_D^A(21)} \tag{7.55}$$

Then the estimated ACR is given by

$$ACR^A = \begin{cases} A(D^A)(M^A) & \text{if } T_D^A \leq 21 \\ A(D^A)(M^A) + \frac{c_R(21) - c_D^A(21)}{G^A} & \text{if } T_D^A > 21 \end{cases} \tag{7.56}$$

where G^A is the average cargo load over all aircraft types found by

$$G^A = \left(N_C^A + N_M^A\right)^{-1} \left[N_C^A \; N_{\mathrm{C-141}} \; N_{\mathrm{C-5}} \; N_{\mathrm{C-141}}\right] \left[G_{\mathrm{CRAF}} \; G_{\mathrm{C-141}} \; G_{\mathrm{C-5}} \; G_{\mathrm{C-17}}\right]' \tag{7.57}$$

Therefore, based on these calculations, we have a method for determining the same output performance measures using the analytical model that are estimated by AFM in the RSM study.

We now consider how the four treatment levels are adjusted in the conceptual analytical model. The four treatment levels in AFM are the number of each of the four aircraft types in the aircraft fleet. For the conceptual analytical model, we can also input the number of aircraft in the fleet. However we have only two inputs, the number of CRAF aircraft, N_C^A, and the number of military aircraft, N_M^A, which is the total of all C-141, C-5, and C-17 aircraft in the AFM scenario, when computing the closed queueing network using the MVA algorithm. Therefore, we can adjust the treatment levels in the conceptual analytical model to match the treatment levels in AFM as illustrated in Table 7.18. The obvious difference, and problem, is that it appears that there are only 8 distinct design points out of the 16 different design points. At this level of analysis, the analytical model doesn't meet the requirement of unique treatment mappings.

Other inputs to the conceptual analytical model, including the formulas for computing the output performance measures, do uniquely determine each of the 16 experimental design points. For example, the realized mean service times and routing proportions from AFM and then input to the conceptual analytical model are determined by the aircraft mix. These different input values will contribute to differentiate each of the experimental design points in the analytical model. More importantly, the mean cargo throughput per hour computed using Equation (7.36) is a function of

Table 7.18 Initial analytical model RSM uncoded 2^4 factorial design.

Design	Number of Aircraft	
Point	CRAF	Military
1	30	190
2	30	170
3	30	170
4	30	150
5	30	170
6	30	150
7	30	150
8	30	130
9	20	190
10	20	170
11	20	170
12	20	150
13	20	170
14	20	150
15	20	150
16	20	130

the different average cargo load for each type of aircraft and the probability that each particular type of military aircraft delivers the cargo. That probability is a function of the actual number of each type of military aircraft as defined by the AFM experimental design point. Hence, the output of the conceptual analytical model is a function of each unique AFM experimental design point and can be adjusted in the same manner as that of AFM.

We have validated the conceptual analytical model to AFM using the conceptual analytical model validation process outlined in Section 6.3.1 and Figure 7.1. We next address surrogate search operational validity by performing the ACV method at each of the initial experimental design points. We again decline to describe the computerized model verification process other than to say we did verify our computerized model. In this case the process was fairly simple since we are using a more simplified version of the same MVA algorithm used in the Psuedo-BRACE application above.

7.3.5.2 Surrogate Search Operational Validity. We determine surrogate search operational validity by using the two-step process described in Section 6.3.3 and outlined in Figure

7.2. The first step is to replicate the AFM simulation model at each of the experimental design points in order to estimate the performance measures using the ACV method. The results of the ACV method are then analyzed to assess the predictive ability of the analytical model. During the second step of the process we estimate response surfaces for both models in order to compare the two models across the entire experimental design space. If both of these steps return satisfactory results the analytical model meets operational validity and can then be used to perform a surrogate search.

We generate 10 replications of AFM at each of the 16 experimental design points in order to initiate the RSM study and perform ACV design point validation. We use the ACV method to estimate the average early accumulated cargo per day per aircraft, ECA, and the aircraft-cargo ratio, ACR. We then analyze the results for each performance measure in turn in accordance with Figure 7.2 to assess ACV design point validity using the criteria:

1. ACV linear regression model is appropriate
   - (a) "Significant" variance reduction
   - (b) Linear scatter plot
2. $\hat{\beta} \approx 1.0$
3. $\bar{Y}^S \approx \bar{Z}$

The variance reduction achieved, $\hat{\beta}$, AFM estimated ECA, $\overline{ECA}^S$, the mean of the ACV, $\overline{ECA}^A$, the relative difference between $\overline{ECA}^S$ and $\overline{ECA}^A$, and the observed variance of $\overline{ECA}^S$ at each of the 16 initial design points are listed in Table 7.19. Since variance reduction is not achieved at all design points, the estimated simulation output reported is not the ACV controlled response. At first glance, it doesn't appear that the analytical model achieves ACV design point validation. At only 9 out of the 16 design points does the ACV actually reduce variance and at only one design point (12) does the level of variance reduction approach a "significant" level. Further, none of the estimated values of β can be considered to be equal to approximately one, and the relative difference between $\overline{ECA}^S$ and $\overline{ECA}^A$ is fairly large. However, further analysis using some of the alternative criteria from section 6.3.3.1 reveals a closer correspondence between the two models than initially thought.

Table 7.19 ACV results for ECA at all design points.

Early Accumulated Cargo per Day per Aircraft						
Design Point	Variance Reduction (%)	$\hat{\beta}$	$\overline{ECA}^S$	$\overline{ECA}^A$	Relative Difference (%)	$Var\left(\overline{ECA}^S\right)$
1	11.90	0.389	30.90	25.01	19.05	0.0005
2	-12.31	-0.037	33.87	27.19	19.72	0.0018
3	-11.46	0.113	34.00	27.19	20.02	0.0017
4	-8.76	0.263	37.49	30.03	19.90	0.0026
5	-6.14	-0.249	34.02	27.28	19.82	0.0015
6	-12.49	-0.008	37.56	30.21	19.52	0.0008
7	1.54	-0.411	37.87	30.13	20.44	0.0042
8	-12.49	-0.012	42.08	33.64	20.04	0.0037
9	9.84	0.405	32.35	26.05	19.47	0.0008
10	-8.83	-0.186	35.61	28.42	20.20	0.0011
11	-5.25	0.456	35.75	28.45	20.43	0.0017
12	28.95	0.479	39.68	31.51	20.58	0.0034
13	-0.48	0.104	35.78	28.70	19.77	0.0004
14	-12.34	0.048	39.70	31.67	20.21	0.0022
15	2.65	0.311	39.98	31.68	20.75	0.0010
16	-11.62	0.137	44.71	35.77	20.01	0.0066

We examine two aspects of the results in Table 7.19 to further assess analytical model validity at the design point level. First of all, the estimated variance of $\overline{ECA}^S$ is extremely small compared to the observed value of $\overline{ECA}^S$ which makes the task of variance reduction very difficult. Given the relatively small amount of observed variance it is not surprising that the ACV is not very successful at reducing variance. Therefore, the results for variance reduction achieved and for the value of $\hat{\beta}$ may not be fair indicators of surrogate performance. Secondly, the relative difference between the means of the two models is approximately identical to 20% at every design point, indicating that although the model outputs are not approximately the same, a very simple adjustment can be made to $\overline{ECA}^A$ in order for the models to be approximately equal. A simple calculation yields that on average $\overline{ECA}^S$ is approximately 1.25 times larger than $\overline{ECA}^A$ at each of the design points. Thus, we can compute an "adjusted" analytical model output, $\overline{ECA}^A_{Adj}$ by

$$\overline{ECA}^A_{Adj} = 1.25\overline{ECA}^A \approx \overline{ECA}^S \tag{7.58}$$

Table 7.20 lists the observed ratio of $\overline{ECA}^S$ to $\overline{ECA}^A$ at each design point, the mean of that ratio, the value of $\overline{ECA}^A_{Adj}$, and the relative difference between $\overline{ECA}^S$ and $\overline{ECA}^A_{Adj}$ at each design point. Obviously, the observed small relative difference between the two models using the adjusted analytical model output indicates that the adjusted analytical model is an excellent predictor of the AFM output for ECA at each design point.

Table 7.20 Adjusted ACV results for ECA at all design points.

Design					Relative
Point	$\overline{ECA}^S$	$\overline{ECA}^A$	$\overline{ECA}^S/\overline{ECA}^A$	$\overline{ECA}^A_{Adj}$	Difference (%)
1	30.90	25.01	1.235	31.27	-1.20
2	33.87	27.19	1.246	33.99	-0.35
3	34.00	27.19	1.250	33.99	0.02
4	37.49	30.03	1.248	37.54	-0.13
5	34.02	27.28	1.247	34.10	-0.23
6	37.56	30.21	1.243	37.76	-0.53
7	37.87	30.13	1.257	37.66	0.55
8	42.08	33.64	1.251	42.05	0.05
9	32.35	26.05	1.242	32.57	-0.66
10	35.61	28.42	1.253	35.52	0.24
11	35.75	28.45	1.257	35.56	0.54
12	39.68	31.51	1.259	39.39	0.72
13	35.78	28.70	1.246	35.88	-0.29
14	39.70	31.67	1.253	39.59	0.26
15	39.98	31.68	1.262	39.61	0.93
16	44.71	35.77	1.250	44.71	0.01
Mean Ratio			1.250		

We also examine the two-dimensional scatter plots between $\overline{ECA}^S$ and $\overline{ECA}^A$ provided in Figure 7.10. The data from four design points are included in each scatter plot. The scatter plots indicate that ECA output from both models are clustered within small neighborhoods of their sample mean. This confirms our observation that there is little variance in $\overline{ECA}^S$ and that if we make an appropriate adjustment, the analytical model provides an excellent prediction of ECA at each of the design points. Therefore, based on the alternative criteria, we consider the first step in surrogate search operational validity to be complete for the estimators of ECA.

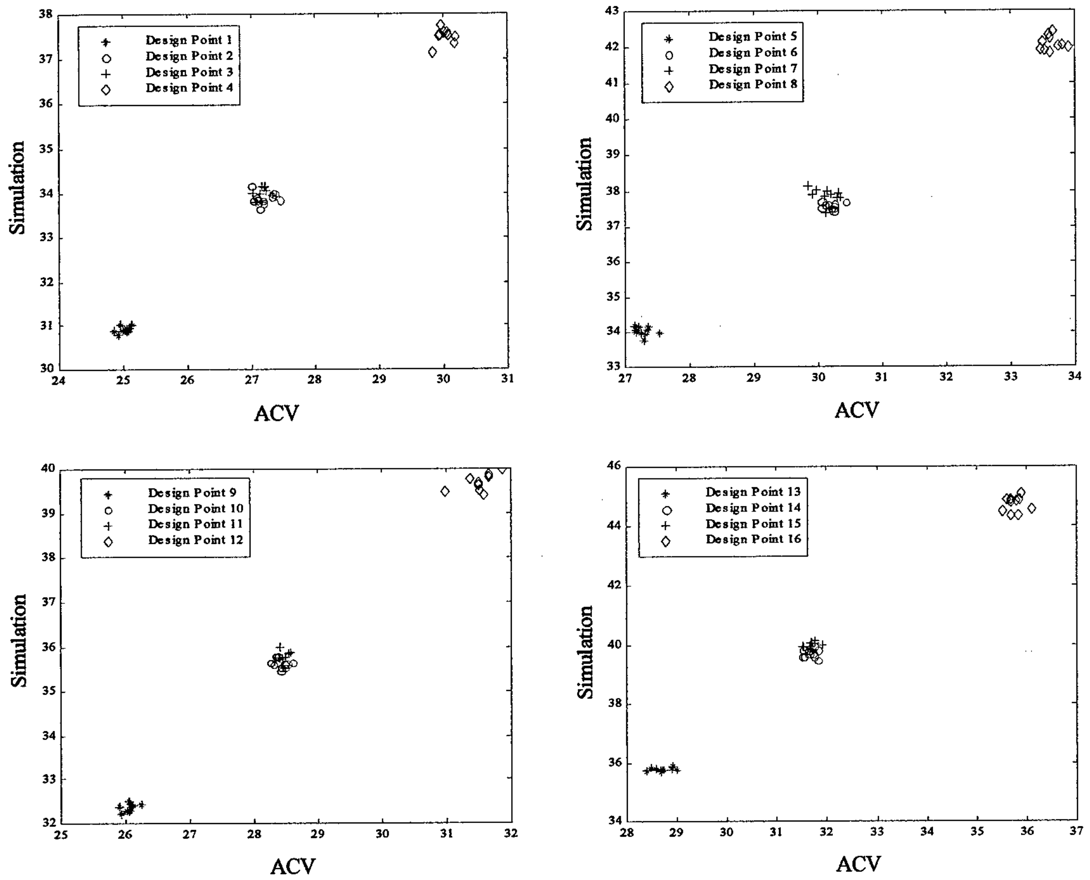

Figure 7.10 Two-way scatter plots for ECA.

We now turn our attention to the aircraft cargo ratio, ACR, results. Table 7.21 contains the observed variance reduction achieved, $\hat{\beta}$, AFM estimated ACR, $\overline{ACR}^S$, the mean of the ACV, $\overline{ACR}^A$, the relative difference between $\overline{ACR}^S$ and $\overline{ACR}^A$, and the observed variance of $\overline{ACR}^S$ at each of the 16 initial design points. Again, we don't meet the normal criteria for ACV design point validation. In this case, variance reduction is achieved at only 4 of the 16 design points and $\hat{\beta}$ is approximately equal to 1 at only 5 different design points. However, the relative difference between $\overline{ACR}^S$ and $\overline{ACR}^A$ is not that large and relatively constant at an average of 3.5 percent.

Table 7.21 ACV results for ACR at all design points.

Aircraft-Cargo Ratio						
Design Point	Variance Reduction (%)	$\hat{\beta}$	$\overline{ACR}^S$	$\overline{ACR}^A$	Relative Difference (%)	$Var\left(\overline{ACR}^S\right)$
1	-11.47	-1.23	178.47	172.05	3.60	0.118
2	-9.83	1.19	162.71	156.63	3.74	0.042
3	26.96	-8.59	162.44	156.64	3.57	0.085
4	-11.96	0.76	146.65	141.20	3.72	0.035
5	-11.53	-1.23	162.05	156.52	3.41	0.121
6	-1.38	-2.52	146.45	141.04	3.70	0.033
7	6.83	-6.83	146.08	141.09	3.42	0.101
8	-11.01	-1.91	130.34	125.61	3.63	0.023
9	29.92	-6.52	170.49	164.35	3.60	0.082
10	-12.49	-0.10	154.24	148.96	3.42	0.027
11	-11.72	0.90	154.15	149.01	3.33	0.063
12	-10.01	0.29	138.40	133.61	3.47	0.059
13	-6.82	2.90	154.39	148.85	3.59	0.078
14	-1.00	3.45	138.61	133.37	3.78	0.033
15	10.48	-9.60	138.05	133.41	3.36	0.112
16	-0.30	-12.09	121.68	117.81	3.18	0.172

As with ECA, we assess the ACV design point validity of the analytical model by considering the alternative criteria. First we note that the observed variance of $\overline{ACR}^S$ is small compared to its observed value making variance reduction very difficult to achieve. So we again disregard the variance reduction results and focus on the relative difference between the two models. As previously noted, the relative difference between the observed means of the two models is fairly constant at 3.5 percent. We adjust $\overline{ACR}^A$ as before by first finding the sample mean over all 16 design points of the ratio of $\overline{ACR}^S$ to $\overline{ACR}^A$ which is approximately equal to 1.037. Then the adjusted analytical model output is given by

$$\overline{ACR}^A_{Adj} = 1.037\overline{ACR}^A \approx \overline{ECA}_S \qquad (7.59)$$

We present the results of the above calculations in Table 7.22 at each design point which indicate that the adjusted analytical model is an excellent predictor for $\overline{ACR}^S$ at each design point.

Table 7.22 Adjusted ACV results for ACR at all design points.

Design Point	$\overline{ACR}^S$	$\overline{ACR}^A$	$\overline{ACR}^S/\overline{ACR}^A$	$\overline{ACR}^A_{Adj}$	Relative Difference (%)
1	178.47	172.05	1.037	178.35	0.07
2	162.71	156.63	1.039	162.37	0.21
3	162.44	156.64	1.037	162.38	0.04
4	146.65	141.20	1.039	146.37	0.19
5	162.05	156.52	1.035	162.25	-0.12
6	146.45	141.04	1.038	146.20	0.17
7	146.08	141.09	1.035	146.25	-0.12
8	130.34	125.61	1.038	130.20	0.10
9	170.49	164.35	1.037	170.36	0.07
10	154.24	148.96	1.035	154.42	-0.12
11	154.15	149.01	1.034	154.47	-0.21
12	138.40	133.61	1.036	138.50	-0.07
13	154.39	148.85	1.037	154.30	0.06
14	138.61	133.37	1.039	138.25	0.25
15	138.05	133.41	1.035	138.30	-0.18
16	121.68	117.81	1.033	122.12	-0.37
Mean Ratio			1.037		

The two-dimensional scatter plots between $\overline{ACR}^A$ and $\overline{ACR}^S$ are presented in Figure 7.11 with data from four design points included in each scatter plot. We observe that the output from both models are clustered within small neighborhoods of their respective sample means as a resulting in the small observed variance for $\overline{ECA}^S$. Further, the plots confirm our claim that by adjusting the analytical model output by Equation (7.59), the analytical model meets the alternative criteria for surrogate search operational validity at the design point level for ACR.

Given that the analytical model has passed the first step of surrogate search operational validity we assess the second step—response surface validity. We begin by estimating the response surface generated by both AFM and the analytical model. The estimated coefficients for the response surfaces for both ECA and ACR are listed in Table 7.23. For the AFM responses we also include their estimated variance. The AFM coefficients are estimated using all 160 uncontrolled

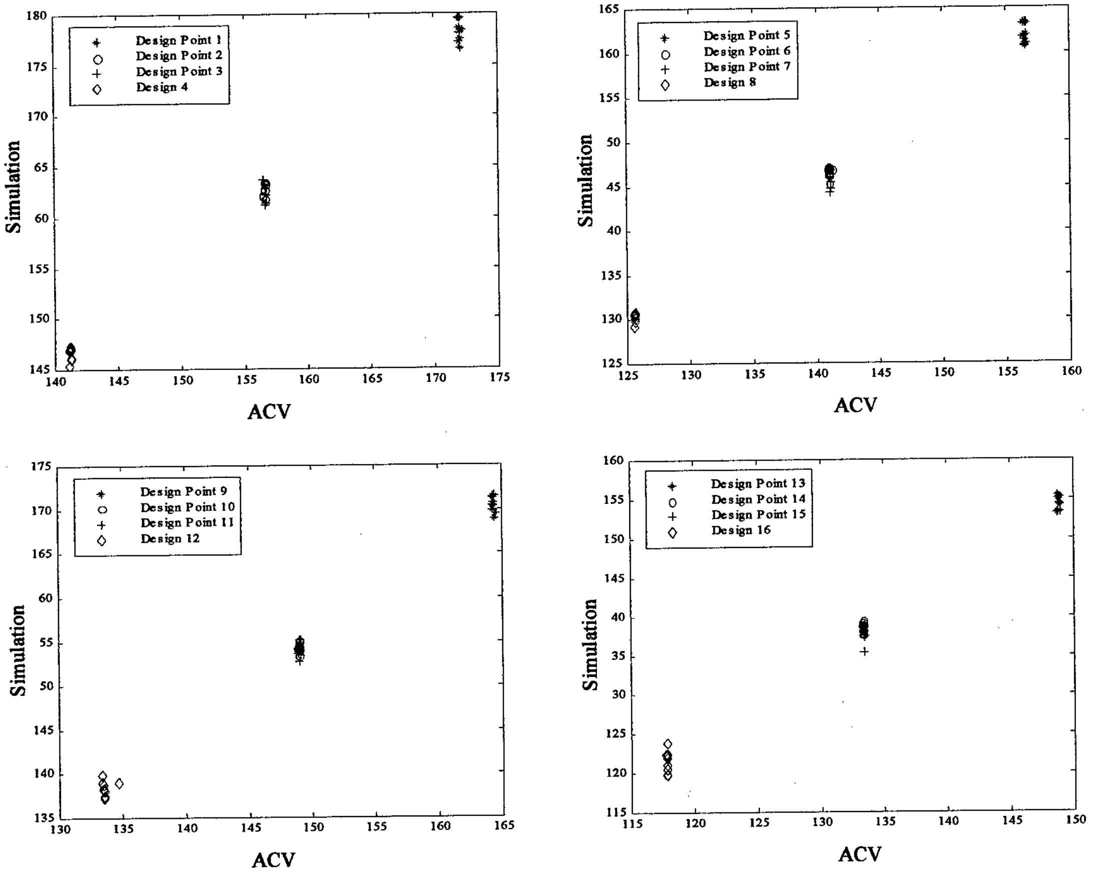

Figure 7.11 Two-way scatter plots for ACR.

responses for both performance measures. The analytical coefficients are estimated by using the least squares approximation method at each of the 16 design points. At each design point, we calculate the analytical responses using the sample means of the appropriate service times and routing probabilities generated by AFM. Then we calculate each analytical response using the adjusted responses as outlined in Equations (7.58) and (7.59).

To determine response surface validity we compare the gradients of each estimated response surface and the estimated responses at the center of the experimental design. We compare the gradient vectors, ∇^S and ∇^A by determining $\cos\psi$ where ψ is the angle formed between the two gradients. If $\cos\psi \approx 1$ then the two gradients point in similar directions indicating that both

Table 7.23 AFM and analytical model response surface parameter estimates.

AFM Estimated Parameters					
Response	$\hat{b}_0^S$	$\hat{b}_1^S$	$\hat{b}_2^S$	$\hat{b}_3^S$	$\hat{b}_4^S$	$\hat{\sigma}_{\hat{b}}^2$
ECA^S	36.96	-0.97	-2.00	-1.99	-1.88	0.00098
ACR^S	150.33	4.07	8.12	8.10	7.94	0.0045
Analytical Model Estimated Parameters						
Response	$\hat{b}_0^S$	$\hat{b}_1^A$	$\hat{b}_2^A$	$\hat{b}_3^A$	$\hat{b}_4^A$	~
ECA^A	36.95	-0.91	-1.97	-1.87	-1.87	~
ACR^A	150.37	3.99	8.08	8.00	8.02	~

response surfaces are oriented approximately the same in the design space. Then if the response at the design center are also approximately equal, the analytical model provides good predictions of AFM across the design space. The observed values of ψ, $\cos\psi$, the estimated responses at the design center, and their relative difference in percentage are detailed in Table 7.24.

Table 7.24 AFM response surface results.

	Gradient Angle		Design Center		
					Relative
Response	ψ	$\cos\psi$	$\hat{b}_0^S$	$\hat{b}_0^A$	Difference (%)
ECA	1.57	0.9996	36.96	36.95	0.020
ACR	0.60	0.9999	150.33	150.37	-0.029

The results of the response surface comparisons listed in Table 7.24 indicate the response surfaces are approximately the same. The gradient angles for both responses are yield cosines nearly equal to one, and the observed design centers both have a relative difference of less than tenth of a percent. Therefore based on these results and the results of the ACV design point comparisons we conclude that the analytical model meets the surrogate search operational validity criteria and can then be used to conduct a surrogate search for the RSM studies. The results of the surrogate search are described below.

7.3.5.3 AFM Surrogate Search Results. We begin the surrogate search procedure by examining the gradient vectors estimated during the response surface validation process. We first find the unit gradient vectors, v^S_{ECA} and v^S_{ACR}, estimated by the AFM responses ECA^S and ACR^S respectively, which are

$$v^S_{ACR} = \begin{bmatrix} -0.2795 \\ -0.5674 \\ -0.5627 \\ -0.5322 \end{bmatrix}, \quad v^S_{ECA} = \begin{bmatrix} 0.2804 \\ 0.5587 \\ 0.5575 \\ 0.5464 \end{bmatrix} \tag{7.60}$$

We observe that $v^S_{ECA} \approx -v^S_{ACR}$ and that the goal is to maximize ECA and to minimize ACR. Thus, to optimize ECA we must take steps in the direction of v^S_{ECA} and to optimize ACR we must take steps in the direction of $-v^S_{ACR}$. Therefore we can search for the optimal values of ECA and ACR by proceeding in a single direction defined by either v^S_{ECA} or $-v^S_{ACR}$.

To determine the actual surrogate search path and step size we begin with two observations. The first observation is that we can only adjust the inputs to the analytical model at discrete levels that correspond to the number of aircraft in the scenario. The second observation is that the gradient terms that correspond to the coded military aircraft inputs, Θ_2, Θ_3, and Θ_4, are all approximately twice the size of the term that corresponds to the coded CRAF aircraft input, Θ_1. Therefore for simplicity we define the surrogate search steepest ascent gradient, g^{ss}, by

$$g^{ss} = \begin{bmatrix} -1.0 \\ -2.0 \\ -2.0 \\ -2.0 \end{bmatrix} \tag{7.61}$$

Now we determine the smallest step size possible that will result in input factors that correspond to integer values for the uncoded aircraft inputs. Since the gradient terms for the military aircraft are twice the size of the gradient term for the CRAF aircraft we want to find the step size that will reduce the number of CRAF aircraft by one for each step. The appropriate coded step size, Δ^{ss}, is found by setting $\Theta_1 = 1$ in equation 7.28, where Θ_1 is the number of CRAF aircraft, and solving for $x_1 = \Delta^{ss}$. Thus the surrogate search step size is given by

$$\frac{\Delta^{ss} - 25}{5} = 1 \Rightarrow \Delta^{ss} = 0.2 \tag{7.62}$$

Based on this coded step size and surrogate search steepest ascent gradient, the coded and uncoded surrogate search points are listed in Table 7.25. For those coded treatment levels that translate to negative aircraft levels, the aircraft level is set to zero.

Table 7.25 Surrogate search steps.

	Treatment Levels							
Design	Coded				Uncoded			
Point	Θ_1	Θ_2	Θ_3	Θ_4	x_1	x_2	x_3	x_4
1	-0.2	-0.4	-0.4	-0.4	24	46	56	46
2	-0.4	-0.8	-0.8	-0.8	23	42	52	42
3	-0.6	-1.2	-1.2	-1.2	22	38	48	38
4	-0.8	-1.6	-1.6	-1.6	21	34	44	34
5	-1.0	-2.0	-2.0	-2.0	20	30	40	30
6	-1.2	-2.4	-2.4	-2.4	19	26	36	26
7	-1.4	-2.8	-2.8	-2.8	18	22	32	22
8	-1.6	-3.2	-3.2	-3.2	17	18	28	18
9	-1.8	-3.6	-3.6	-3.6	16	14	24	14
10	-2.0	-4.0	-4.0	-4.0	15	10	20	10
11	-2.2	-4.4	-4.4	-4.4	14	6	16	6
12	-2.4	-4.8	-4.8	-4.8	13	2	12	2
13	-2.6	-5.2	-5.2	-5.2	12	0	8	0
14	-2.8	-5.6	-5.6	-5.6	11	0	4	0
15	-3.0	-6.0	-6.0	-6.0	10	0	0	0
16	-3.2	-6.4	-6.4	-6.4	9	0	0	0
17	-3.4	-6.8	-6.8	-6.8	8	0	0	0
18	-3.6	-7.2	-7.2	-7.2	7	0	0	0
19	-3.8	-7.6	-7.6	-7.6	6	0	0	0
20	-4.0	-8.0	-8.0	-8.0	5	0	0	0

We are now faced with the problem of determining the inputs to the analytical model at each of the steps along the path of steepest ascent (descent) listed in Table 7.25. The method for translating the aircraft fleet to the appropriate settings for the analytical model as described above in Section 7.3.5.1 still holds for the surrogate search. The problem is determining the appropriate settings for the aircraft routing probabilities and the service (waiting) times at each of the air bases. For the purposes of generating an ACV the observed sample means from the AFM replications for each of the described values are used as inputs to the analytical model. When performing a surrogate search, these AFM outputs are obviously not available. To set these values, we have essentially two choices. The first choice is to use statistical methods to predict the input values based on the observed values generated during the initial stages of the RSM study. The second choice is base the settings on an understanding of the processes that give rise to the value. We use a combination of both methods as described below.

The analytical model inputs are defined in the following manner. Let the probability transition matrices of aircraft routings be represented by

$$\Pi_Y^A = \begin{bmatrix} 0 & \pi_{Y(1,2)}^A & \cdots & \pi_{(1,13)}^A \\ \pi_{Y(2,1)}^A & 0 & \cdots & \pi_{Y(2,13)}^A \\ \vdots & \vdots & & \vdots \\ \pi_{Y(13,1)}^A & \pi_{Y(13,2)}^A & \cdots & 0 \end{bmatrix}, \quad Y \in \{C,\ M\} \tag{7.63}$$

where $\pi_{Y(i,j)}^A$ is the steady state probability that an aircraft of type Y will fly from base i to base j with $Y \in \{C,\ M\}$ representing CRAF and military aircraft respectively. Since aircraft don't take-off and land at the same airfield, $\pi_{Y(i,i)}^A = 0$ for $i = 1, 2, \ldots, 13$. Note that these 13 × 13 matrices are transformed into matrices of size up to 169 to represent the total state space of the model, however, each Π_Y^A completely determines the size and values of each of the larger matrices. Therefore, we consider only Π_C^A and Π_M^A for this discussion. We let the mean time that an aircraft of type Y spends at airbase i, $i = 1, 2, \ldots, 13$ by $W_{Y(i)}^A$. As before, the number of each type of aircraft are symbolized by N_C^A for CRAF aircraft and $N_M^A = N_{C-141} + N_{C-5} + N_{C-17}$ for military aircraft with N_{C-141}, N_{C-5}, and N_{C-17} the number of C-141, C-5, and C-17 aircraft respectively at the particular design point.

In a similar manner, we define the corresponding terms for AFM generated values. To define the observed routing probabilities for AFM let $t_{Y(i,j)}(k)$ be the number of aircraft of type Y that fly from airbase i to airbase j $(i, j = 1, 2, \ldots, 13)$ during AFM replication $k = 1, 2, \ldots, n$. Again, $t_{Y(i,i)} = 0$ for $i = 1, 2, \ldots, 13$. The total number of departures observed at base i for aircraft type Y during replication k is then given by

$$T_{Y(i)}(k) = \sum_{j=1}^{13} t_{Y(i,j)}(k), \quad i = 1, 2, \ldots, 13 \;\; k = 1, 2, \ldots, n \tag{7.64}$$

so that the routing proportions for replication k are given by

$$\pi^S_{Y(i,j)}(k) = \frac{t_{Y(i,j)}(k)}{T_{Y(i)}(k)} \quad i, j = 1, 2, \ldots, 13 \;\; k = 1, 2, \ldots, n \;\; Y \in \{C,\ M\} \tag{7.65}$$

with the estimated routing probabilities computed as

$$\bar{\pi}^S_{Y(i,j)} = n^{-1} \sum_{k=1}^{n} \pi^S_{Y(i,j)}, \quad i, j = 1, 2, \ldots, 13 \tag{7.66}$$

The two routing probability matrices are then given by

$$\Pi^S_Y = \begin{bmatrix} 0 & \bar{\pi}^S_{Y(1,2)} & \cdots & \bar{\pi}^S_{(1,13)} \\ \bar{\pi}^S_{Y(2,1)} & 0 & \cdots & \bar{\pi}^S_{Y(2,13)} \\ \vdots & \vdots & & \vdots \\ \bar{\pi}^S_{Y(13,1)} & \bar{\pi}^S_{Y(13,2)} & \cdots & 0 \end{bmatrix}, \quad Y \in \{C,\ M\} \tag{7.67}$$

To estimate the mean time aircraft spend at each base during each visit, we let $w_{Y(i,l)}(k)$ be the amount of time the l^{th} departing aircraft of type Y spends at base i during replication k.

Thus, the mean time an aircraft of type Y spends at base i during a single visit for replication k is computed as

$$W_{Y(i)}^S(k) = \frac{1}{T_{Y(i)}(k)} \sum_{l=1}^{T_{Y(i)}(k)} w_{Y(i,l)}(k), \quad i = 1, 2, \ldots, 13 \quad k = 1, 2, \ldots, n \; Y \in \{C,\ M\} \tag{7.68}$$

so that we can estimate the mean time aircraft of each type spend at each base by

$$\overline{W}_{Y(i)}^S = n^{-1} \sum_{k=1}^{n} W_{Y(i)}^S(k), \quad i = 1, 2, \ldots, 13 \quad Y \in \{C,\ M\} \tag{7.69}$$

For the replications generated during the initial first order design of experiment, we observe that each of the values of Π_Y^S and $\overline{\mathbf{W}}_Y^S = (\overline{W}_{Y(1)}^S, \overline{W}_{Y(2)}^S, \ldots, \overline{W}_{Y(13)}^S)'$ at each of the 16 design points possess little variance, except for some of the waiting times. The air bases that exhibit a large variance in waiting times are the three "home stations" of Charleston AFB, McGuire AFB, and JFK IAP. Therefore, we will use the "overall" sample means of Π_Y^S and $\overline{\mathbf{W}}_Y^S$ (except for the three home stations) as inputs to the analytical model for the purposes of performing a surrogate search.

The overall sample means that are used as inputs to the analytical model for the purposes of performing the current surrogate search are defined in the following manner. Let $\Pi_Y^S(d)$ be the estimated routing probability matrices for aircraft type Y at design point $d = 1, 2, \ldots, 16$, where

$$\Pi_Y^S(d) = \left\{ \bar{\pi}_{Y(i,j)}^S(d) \right\}_{\substack{i=1,2\ldots,16 \\ j=1,2,\ldots,16}} \quad d = 1, 2, \ldots, 16 \quad Y \in \{C,\ M\} \tag{7.70}$$

Then the overall sample mean for the routing probability matrices are given by

$$\overline{\Pi}_Y^S = \left\{ \frac{1}{16} \sum_{d=1}^{16} \bar{\pi}_{Y(i,j)}^S(d) \right\}_{\substack{i=1,2\ldots,16 \\ j=1,2,\ldots,16}} \quad Y \in \{C,\ M\} \tag{7.71}$$

Similarly, let $\overline{\mathbf{W}}_Y^S(d)$ be the vector of estimated mean waiting times at air bases $4, 5, \ldots, 13$ (where Charleston AFB is base 1, McGuire AFB is base 2, and JFK IAP is base 3) for aircraft of type Y at initial design point d. The overall sample mean for the waiting time vector is then

$$\overline{\overline{\mathbf{W}}}_Y^S = \left\{ \frac{1}{16}\sum_{d=1}^{16} W_{Y(4)}^S, \frac{1}{16}\sum_{d=1}^{16} W_{Y(5)}^S, \ldots, \frac{1}{16}\sum_{d=1}^{16} W_{Y(13)}^S \right\} \quad Y \in \{C,\ M\} \qquad (7.72)$$

In order to determine the waiting time inputs for the home stations we recall that aircraft wait at their home station for two reasons. First, they must wait in a FIFO queue for an available cargo load. Then if they are assigned a load, they might be grounded a designated amount of time so that the target utilization rate for their type of aircraft is maintained by the differential or capping procedures within AFM [11]. We can compute the grounding time assigned each type of aircraft using the differential use rate control formula used by the differential procedure in AFM in order to estimate the mean grounding times. However, if aircraft must wait an additional amount of time for a mission, this formula will not provide an accurate estimate. Analysis of the initial design region indicates that more aircraft than are required are included in the aircraft fleet and as such, the waiting times at each of the home stations exceed the grounding times computed using the differential use rate control formula. Therefore, we use a two-fold strategy for estimating the waiting times at these bases. First we use linear regression at the 16 design points with the aircraft fleet as the independent variables and each air base waiting time as the dependent variables. We then apply the estimated response functions for each base using the aircraft fleet at the appropriate surrogate search design point. We also compute grounding times based on the differential procedure formula and the aircraft fleet input in order to obtain another estimate of waiting times at each base. If the waiting time computed using the linear regression approach is greater (indicating aircraft must wait for a mission) we use that input, otherwise we use the waiting time computed using the differential use rate formula.

The home station waiting time, $W^A_{Y(i)}(\delta)$, for aircraft of type Y at air bases $i = 1,2,3$ estimated using the differential use rate formula is computed in the following manner. Define an aircraft cycle as all the ground and flying activity that occurs to an aircraft once it is assigned a cargo load until it returns to its home station and is ready to accept a new mission. Then let g_y be the expected sum of all ground times (servicing, cargo up-load and off-load, taxiing, etc.) in hours experienced by an aircraft of type $y \in \{CRAF, C-141, C-5, C-17\}$ during a typical cycle. Further let f_y be the expected sum all flying time, in hours, accumulated during a typical cycle experienced by aircraft of type $y \in \{CRAF, C-141, C-5, C-17\}$. Then the expected utilization rate (without the use of the differential procedure) is given by

$$U_y = \frac{24 f_y}{f_y + g_y} \quad y \in \{CRAF, C-141, C-5, C-17\} \tag{7.73}$$

The expected additional ground time, S_y, or slack time, added to the end of each cycle for aircraft of type y as calculated by the differential use rate formula [11] is given by

$$S_y = \frac{(U_y - \Upsilon_y)(f_y + g_y)^2}{(24 f_y)} \quad y \in \{CRAF, C-141, C-5, C-17\} \tag{7.74}$$

where Υ_y is the input capability rate for each type of aircraft listed in Table 7.16. Using observed values of $\overline{\overline{\mathbf{W}}}^S_Y$ and $\overline{\Pi}^S_Y$ and the standard servicing times listed in Table 7.16, the estimated slack times for each type of aircraft are listed in Table 7.26.

Table 7.26 Expected slack times using differential use rate formula.

	Inputs (hrs)			Slack (hrs)
Aircraft	$E[f]$	$E[g]$	Υ	S
CRAF	29.12	10.75	12.00	12.56
C-141	33.40	10.13	12.20	14.69
C-5	33.19	15.75	10.70	16.77
C-17	31.80	10.13	15.30	6.69

The next step in computing the expected waiting time at each home station, $W^A_{Y(i)}(\delta)$ is to condition on the probability an aircraft experiences its slack time at base i and the probability it on-loads cargo at the base i. We begin by first finding the expected waiting time for military aircraft at bases 1 and 2 (Charleston AFB and McGuire AFB). Let $h_{z(i)}$ be the number of military aircraft of type $z \in Mil = \{C-141, C-5, C-17\}$ assigned base $i = 1,2$ as their home station. We compute $p_{z(i)}$, the probability that a military aircraft of type z has home station i by

$$p_{z(i)} = \frac{h_{z(i)}}{\sum_{z \,\in\, Mil} N_z} \quad i = 1,2 \tag{7.75}$$

where N_z is the number of military aircraft of type z in the aircraft fleet. Next we compute the probability that a military aircraft at station i is a home station aircraft represented by P_i by

$$P_i = \frac{\mathbf{p}_i'\mathbf{N_z}}{\mathbf{p}_i'\mathbf{N_z} + \bar{\bar{\pi}}_{M(i,j)}\mathbf{p}_j'\mathbf{N_z}} \quad i = 1,2 \;\; j = 1,2 \;\; i \neq j \tag{7.76}$$

where $\mathbf{p}_i = \left[p_{C-141(i)}, p_{C-5(i)}, p_{C-17(i)}\right]'$ and $\mathbf{N}_z = [N_{C-141}, N_{C-5}, N_{C-17}]'$. Then by conditioning on the probability that an aircraft at station i is at its home station, the expected waiting time based on the differential rate use formula at military home station i is given by

$$W^A_{M(i)}(\delta) = P_i \left\{\mathbf{p}_i' \left[\mathbf{S}_z + (1 - \bar{\bar{\pi}}_{M(i,j)})\mathbf{L}_z\right]\right\} + (1 - P_i)\left\{\bar{\bar{\pi}}_{M(j,i)}\mathbf{p}_j'\mathbf{L}_z\right\} \quad i = 1,2 \;\; j = 1,2 \;\; i \neq j \tag{7.77}$$

where $\mathbf{S}_z = [S_{C-141}, S_{C-5}, S_{C_17}]'$ is the vector of computed slack times shown in Table 7.26 and $\mathbf{L}_z = [L_{C-141}, L_{C-5}, L_{C-17}]'$ is the vector of standard cargo on-load times from Table 7.16. In a similar manner, we compute the expected waiting time for CRAF aircraft by

$$W^A_{C(3)}(\delta) = \mathbf{S} + (1 - \bar{\bar{\pi}}_{C(3,1)} - \bar{\bar{\pi}}_{C(3,2)})L_C \tag{7.78}$$

Given the methods described above for determining the appropriate inputs to the analytical model, a surrogate search is performed as defined in Table 7.25. The results for both ECA and ACR of the surrogate search are provided in Table 7.27 using ECA^A_{adj} and ACR^A_{adj}. For brevity only the results from the first 10 surrogate search steps are reported. The surrogate search results also are presented graphically in Figures 7.12 and 7.13. The surrogate search produces local optima for both performance measures as indicated in the tables and graphs.

Table 7.27 Surrogate search results.

	Aircraft Levels				Outputs	
Step	CRAF	C-141	C-5	C-17	ECA^A_{adj}	ACR^A_{adj}
1	24	46	56	46	43.39	139.69
2	23	42	52	42	46.42	129.27
3	22	38	48	38	49.93	118.86
4	21	34	44	34	51.15	111.26
5	20	30	40	30	47.89	107.08
6	19	26	36	26	38.06	104.06
7	18	22	32	22	19.22	156.63
8	17	18	28	18	-8.62	246.08
9	16	14	24	14	-63.03	346.26
10	15	10	20	10	-169.34	448.91

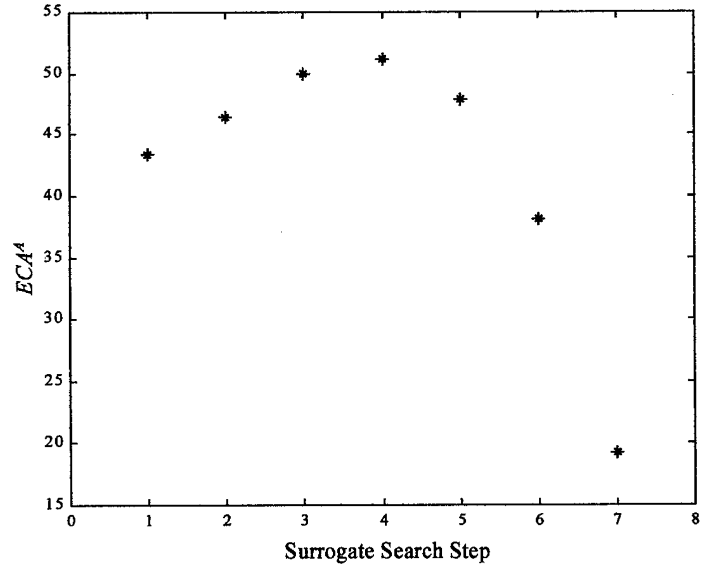

Figure 7.12 Initial surrogate search for ECA.

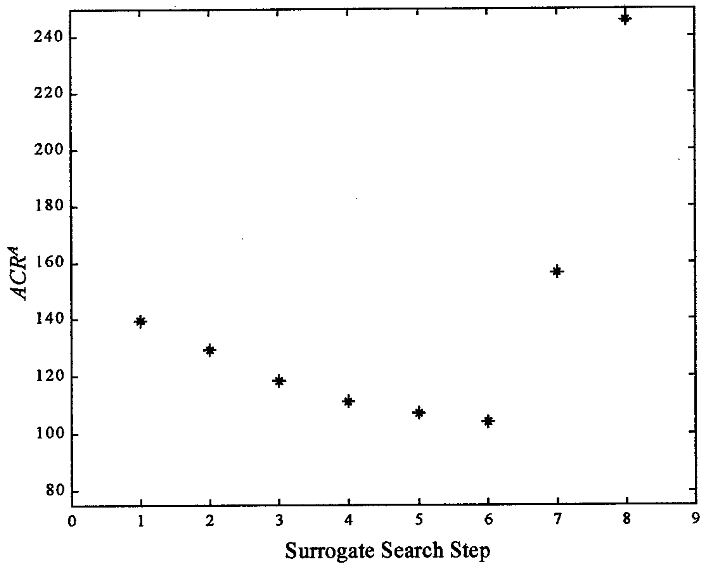

Figure 7.13 Initial surrogate search for ACR.

The next step in the surrogate search procedure is to validate the surrogate search results using AFM replications at what will presumably be design points in the next series of experiments—if the surrogate search results are valid. Due to the number of factors (four) and the limited nature of the surrogate search results from a single gradient direction, it is not clear if the observed local optima (if they are predicted correctly) are anywhere near the global optima that we are searching for. Furthermore, little information is available for determining an appropriate range for each of the treatment levels. Of course this is the same situation we would be in if we didn't use the surrogate search method and had used AFM exclusively to explore the defined gradient direction. The difference is that with the surrogate search methodology, we can rapidly perform additional surrogate searches to gain additional information about the predicted behavior of AFM in the investigated region. Therefore we perform additional surrogate searches along 13 different gradients. These additional gradients point in the same general direction of the original gradient—reducing each aircraft level—however the proportion of aircraft reduction differs for each gradient. The coded gradients for the additional surrogate searches are provided in Table 7.28.

Table 7.28 Additional surrogate search gradients.

	Coded Treatment			
Gradient	Θ_1	Θ_2	Θ_3	Θ_4
1	-1.5	-1.0	-1.0	-1.0
2	-1.0	-1.0	-1.0	-1.0
3	-1.0	-1.5	-1.5	-1.5
4	-1.0	-2.0	-2.0	-2.0
5	-1.0	-2.0	-1.0	-1.0
6	-1.0	-1.0	-2.0	-1.0
7	-1.0	-1.0	-1.0	-2.0
8	-2.0	-2.0	-1.0	-1.0
9	-2.0	-1.0	-2.0	-1.0
10	-2.0	-1.0	-1.0	-2.0
11	-1.0	-2.0	-2.0	-1.0
12	-1.0	-2.0	-1.0	-2.0
13	-1.0	-1.0	-2.0	-2.0

We perform the additional surrogate searches and provide the results in tables 7.29 and 7.30. We list the 10 largest values observed for ECA^A_{Adj} and 10 smallest values observed for ACR^A_{Adj}, over the 14 surrogate search gradients, with the aircraft levels that generated them. The sample means for the aircraft inputs and the performance measures are also included in the tables. The additional gradients have identified additional aircraft levels that produce better performance measure responses than the initial surrogate search. Even more importantly, we have a better idea of the predicted range of each aircraft level that produce desirable performance measure responses—as long as the surrogate search results are valid.

Given the results of the surrogate searches we attempt to validate the results using AFM replications. Rather than randomly choosing some of the observed surrogate search results to validate, we first design a new set of experiments to estimate the next response surface suggested by the surrogate search results. We then validate the surrogate search by replicating the AFM model at several of the proposed experimental design points. If the results are validated, we complete the proposed design using the results of the validation steps.

We begin to design the new set of experiments by first recognizing the need for a design that can estimate a second order response surface and by setting the high, center, and low levels of

Table 7.29 Largest ECA^A_{Adj} observations over all surrogate searches (with sample means).

Aircraft Levels				
CRAF	C-141	C-5	C-17	ECA^A_{Adj}
19	26	48	38	53.92
18	22	46	36	53.60
15	30	50	40	53.10
17	18	44	34	52.71
20	30	50	30	52.56
13	26	48	38	52.09
20	30	50	40	52.08
20	30	40	40	51.62
21	34	44	42	51.57
21	34	52	34	51.51
20	35	45	35	51.24
21	34	44	34	51.15
18	36	46	36	51.14
16	38	48	38	50.88
20	40	50	30	50.83
Mean Values				
19	31	47	37	52.00

each of the aircraft treatments. We also recognize that the aircraft level ranges that are suggested by the surrogate search results for the two performance measures are slightly different. However, we choose to design a single set of experiments to estimate response surfaces for both ECA^S and ACR^S. Therefore based on the surrogate search results we decide to set the design center at $x_1^c = 15$, $x_2^c = 25$, $x_3^c = 40$, and $x_4^c = 30$. The low level for each treatment is given by $x_1^L = 5$, $x_2^L = 5$, $x_3^L = 25$ and $x_4^L = 15$ and the high levels as $x_1^H = 25$, $x_2^H = 45$, $x_3^H = 55$ and $x_4^H = 45$. Thus, the coded treatment levels for the proposed design of experiment are given by

$$\Theta_1^j = \frac{x_1^j - 15}{10}, \quad j = c, H, L \tag{7.79}$$

$$\Theta_2^j = \frac{x_2^j - 25}{20}, \quad j = c, H, L \tag{7.80}$$

$$\Theta_3^j = \frac{x_3^j - 40}{15}, \quad j = c, H, L \tag{7.81}$$

$$\Theta_4^j = \frac{x_4^j - 30}{15}, \quad j = c, H, L \tag{7.82}$$

Table 7.30 Smallest ACR^A_{Adj} observations over all surrogate searches (with sample means).

Aircraft Levels				
CRAF	C-141	C-5	C-17	ACR^A_{Adj}
15	10	40	30	92.28
7	14	42	32	92.76
14	6	38	28	94.10
16	14	42	32	95.60
9	18	44	34	96.90
17	18	28	34	98.03
17	18	44	18	98.56
17	18	44	34	99.27
18	22	32	36	101.49
11	22	46	36	101.79
18	22	46	22	101.95
13	26	36	26	102.05
9	28	38	28	102.70
5	30	40	30	103.81
14	28	38	28	103.84
Mean Values				
14	20	40	30	99.01

In most cases, Box and Draper recommend a central composite design (CCD) for estimating a second order response surface [12]. A CCD is a full factorial design augmented with center point experiments and a number of "axial" or "star" coded design points of the form $(\pm\alpha, 0, \ldots, 0), (0, \pm\alpha, \ldots, 0), \ldots, (0, 0, \ldots, \pm\alpha)$ where α is usually chosen equal to $(n_f)^{1/4}$ with n_f the number of factorial points in the design. In our case then, $n_f = 16$ so that $\alpha = 2$. This presents a difficulty though because $\Theta_1^L = -2$ and $\Theta_2^L = -2$ translate to CRAF and C-141 aircraft levels of -5 and -15 which are of course physically impossible. Therefore we decide to construct a *Box-Behnken* design that requires only the three levels already defined for each treatment in order to estimate a second order response surface [12]. Although not all Box-Behnken designs are rotatable, the Box-Behnken design for 4 variables is a rotatable second order design that requires only 27 trials [12]. The proposed experimental design appears in Table 7.31.

We now attempt to validate the surrogate search results by performing AFM replications. Examination of the proposed Box-Behnken design of experiment and the surrogate searches reveals

Table 7.31 Proposed second order Box-Behnken design of experiment.

	Coded				Uncoded			
Design	Treatment Levels				Treatment Levels			
Point	Θ_1	Θ_2	Θ_3	Θ_4	CRAF	C-141	C-5	C-17
1	-1	-1	0	0	5	5	40	30
2	1	-1	0	0	25	5	40	30
3	-1	1	0	0	5	45	40	30
4	1	1	0	0	25	45	40	30
5	0	0	-1	-1	15	25	25	15
6	0	0	1	-1	15	25	55	15
7	0	0	-1	1	15	25	25	45
8	0	0	1	1	15	25	55	45
9	0	0	0	0	15	25	40	30
10	-1	0	0	-1	5	25	40	15
11	1	0	0	-1	25	25	40	15
12	-1	0	0	1	5	25	40	45
13	1	0	0	1	25	25	40	45
14	0	-1	-1	0	15	5	25	30
15	0	1	-1	0	15	45	25	30
16	0	-1	1	0	15	5	55	30
17	0	1	1	0	15	45	55	30
18	0	0	0	0	15	25	40	30
19	0	-1	0	-1	15	5	40	15
20	0	1	0	-1	15	45	40	15
21	0	-1	0	1	15	5	40	45
22	0	1	0	1	15	45	40	45
23	-1	0	-1	0	5	25	25	30
24	1	0	-1	0	25	25	25	30
25	-1	0	1	0	5	25	55	30
26	1	0	1	0	25	25	55	30
27	0	0	0	0	15	25	40	30

that none of the proposed design points are evaluated during the surrogate search. Since the analytical model can be computed very rapidly, we simply pick 3 of the 27 possible design points and evaluate the analytical model at those points—using the surrogate search inputs—as well as performing 10 independent replications of AFM at the same design points. Design points 5 ($\Theta = [0, 0, -1, -1]'$), 9 ($\Theta = [0, 0, 0, 0]'$), and 8 ($\Theta = [0, 0, 1, 1]'$) are chosen for validation. The results, listed in Table 7.32, indicate that the surrogate analytical model is an adequate predictor of AFM output behavior at the tested design points. Based on these results, we decide to rely on the surrogate search results and perform the proposed Box-Behnken design of experiment.

Table 7.32 Surrogate search validation results.

Design Point	Aircraft Levels				ECA		ACR	
	CRAF	C-141	C-5	C-17	Surrogate	AFM	Surrogate	AFM
5	15	25	25	15	-29.43	-13.66	317.60	300.62
9	15	25	40	30	45.29	46.78	101.91	94.88
8	15	25	55	45	52.34	47.67	113.96	113.78

To estimate second order response surfaces for both of the performance measures, we generate 10 replications of AFM at the remaining 24 design points defined in Table 7.31. Based on these replications we estimate response surfaces of the form

$$\begin{aligned} E[y] \;=\; & \hat{b}_0 + \hat{b}_1\Theta_1 + \hat{b}_2\Theta_2 + \hat{b}_3\Theta_3 + \hat{b}_4\Theta_4 + \\ & \hat{b}_{11}\Theta_1^2 + \hat{b}_{22}\Theta_2^2 + \hat{b}_{33}\Theta_3^2 + \hat{b}_{44}\Theta_4^2 + \\ & \hat{b}_{12}\Theta_1\Theta_2 + \hat{b}_{13}\Theta_1\Theta_3 + \hat{b}_{14}\Theta_1\Theta_4 + \\ & \hat{b}_{23}\Theta_2\Theta_3 + \hat{b}_{24}\Theta_2\Theta_4 + \\ & \hat{b}_{34}\Theta_3\Theta_4 \end{aligned} \tag{7.83}$$

We estimate the response surfaces without the use of an ACV due to the small observed variance of each performance measure response in the simulation results. The estimated parameters and their associated standard error are listed in Table 7.33. For the ECA^S response surface, $MSE = 1.78$, $R^2 = 0.99$, and the regression F statistic is equal to 2964.6. The same values for the ACR^S response surface are $MSE = 253.67$, $R^2 = 0.89$, and the F statistic is 142.5. Thus, in both cases we accept the fitted response surfaces as statistically adequate.

We now analyze the fitted response surfaces to determine the optima or identify a new space for further experimentation. First we note that the maximum observed $\overline{ECA}^S = 58.58$ at design point 21 (CRAF = 15, C-141 = 5, C-5 = 40, and C-17 = 45) and the minimum observed $\overline{ACR}^S =$ 91.44 also at design point 21. We now attempt to locate the stationary points of each surface, if

Table 7.33 Second order response surfaces parameter estimates.

Parameter	ECA^S Parameter Estimate	ECA^S Standard Error	ACR^S Parameter Estimate	ACR^S Standard Error
$\hat{b}_0$	47.10	0.24	94.88	2.91
$\hat{b}_1$	2.86	0.12	-5.82	1.45
$\hat{b}_2$	2.31	0.12	-7.24	1.45
$\hat{b}_3$	13.22	0.12	-27.54	1.45
$\hat{b}_4$	15.47	0.12	31.80	1.45
$\hat{b}_{11}$	-2.62	0.18	0.94	2.18
$\hat{b}_{22}$	-1.90	0.18	5.56	2.18
$\hat{b}_{33}$	-6.63	0.18	24.95	2.18
$\hat{b}_{44}$	-8.39	0.18	29.88	2.18
$\hat{b}_{12}$	-2.11	0.21	0.86	2.52
$\hat{b}_{13}$	-4.57	0.21	15.81	2.52
$\hat{b}_{14}$	-5.34	0.21	20.76	2.52
$\hat{b}_{23}$	-8.63	0.21	27.12	2.52
$\hat{b}_{24}$	-9.31	0.21	25.94	2.52
$\hat{b}_{34}$	-15.13	0.21	56.13	2.52

they exist. Recall that the stationary point of a second order function with 4 variables is given by

$$\Theta_s = -\frac{1}{2}\mathbf{B}^{-1}\mathbf{b} \tag{7.84}$$

where $\mathbf{b} = (b_1, b_2, b_3, b_4)'$ and

$$\mathbf{B} = \begin{bmatrix} b_{11} & \frac{1}{2}b_{12} & \frac{1}{2}b_{13} & \frac{1}{2}b_{14} \\ \frac{1}{2}b_{12} & b_{22} & \frac{1}{2}b_{23} & \frac{1}{2}b_{24} \\ \frac{1}{2}b_{13} & \frac{1}{2}b_{23} & b_{33} & \frac{1}{2}b_{34} \\ \frac{1}{2}b_{14} & \frac{1}{2}b_{24} & \frac{1}{2}b_{34} & b_{44} \end{bmatrix} \tag{7.85}$$

Then the estimated stationary points for the AFM fitted response surfaces are

$$\Theta_s^{ECA} = \begin{bmatrix} 0.89 \\ 4.83 \\ 4.57 \\ -6.16 \end{bmatrix}, \quad \Theta_s^{ACR} = \begin{bmatrix} 0.78 \\ 0.15 \\ -0.05 \\ 0.24 \end{bmatrix} \tag{7.86}$$

where Θ_s^{ECA} is the estimated stationary point for the ECA^S response surface and Θ_s^{ACR} is the estimated stationary point for the ACR^S response surface. For ECA, the stationary point is approximately 9.11 coded units from the design center, which is far outside of the experimental design region. On the other hand, the stationary point for the ACR response surface is only 0.84 coded units from the design center—well within the design region. We now examine the eigenvalues of **B**, which is equal to one half of the Hessian matrix, to determine the nature of the stationary points. The eigenvalues for each response surface are

$$\boldsymbol{\lambda}^{ECA} = \begin{bmatrix} 0.72 \\ 0.09 \\ -1.83 \\ -18.52 \end{bmatrix}, \quad \boldsymbol{\lambda}^{ACR} = \begin{bmatrix} 4.75 \\ 1.47 \\ -3.27 \\ -64.28 \end{bmatrix} \tag{7.87}$$

indicating that both stationary points are saddle points, not global optima. Hence canonical analysis or ridge analysis is required to identify the direction for further experimentation, if required.

We use ridge analysis to identify promising points for further investigation of the experimental design region. Limited ridge analysis results, computed by the SAS statistical computer program, are listed in Table 7.34. The table includes the coded distance from the design center, estimated response, standard error of the estimated response, and aircraft levels for 4 selected points for each performance measure. We performed AFM replications at each of these design points to validate the ridge analysis and have provided the resulting estimated responses at those points. For both measures, improvements in the levels of each response are observed until the last point replicated.

Table 7.34 Ridge analysis.

ECA Ridge Analysis							
Coded	Estimated	Standard	Aircraft Level				AFM
Radius	Response	Error	CRAF	C-141	C-5	C-17	Response
0.5	53.88	0.22	15	22	45	36	53.00
1.1	58.60	0.23	14	11	48	40	56.53
1.7	63.31	0.46	13	0	51	43	58.64
2.5	70.22	1.04	12	0	55	46	56.72
ACR Ridge Analysis							
1.0	77.82	2.62	10	14	46	38	88.08
1.6	66.80	4.84	5	7	48	41	88.51
2.3	49.71	10.39	0	0	50	44	76.64
2.0	28.00	18.35	0	0	52	47	80.04

The AFM ridge analysis results indicate that we just missed designing a second order design of experiment that would have contained the feasibly optimal results. In particular, if we had set the low levels for CRAF and C-141 aircraft to zero instead of 5, the "best" ridge analysis results would have been in the design. Overall, the results obtained with our second order design are certainly superior compared to the results of what would have likely been the second order design if we had not used the surrogate search method. It is unlikely that an analyst would have set CRAF and C-141 aircraft as low as we did based on a AFM steepest ascent search. Additionally, for ECA it appears that there is actually little improvement outside the actual design region since the maximum observed $\overline{ECA}^S = 58.58$ within the original second order design statistically equivalent to the maximum observed along the ridge analysis (58.64). For ACR, it appears there is a real improvement to the observed values outside of the second order design region. Of course these results are only important if AMC planner are interested in composing air lift fleets without CRAF and/or C-141 aircraft. If desired additional experiments could be performed to further define the response surfaces to include these new points.

In summary, we have demonstrated the effectiveness of the surrogate search method when applied to a "real-world" simulation model and problem. Despite the many problems in applying the method to AFM and this particular RSM study, namely small variance for the performance measures and the defining of appropriate analytical model inputs and outputs, the surrogate search method was quite effective in defining the new second order design region. Although the ACV method was not efficient in reducing study times at the initial first order stage of the study, the additional effort in validating the analytical model paid off in the end. The information about the predicted behavior of AFM gained through the performance of numerous surrogate searches provided us with a second order design region that nearly encompassed the entire range of "interesting" AFM output. Without that additional surrogate search knowledge it is unlikely that an analyst would have designed a set of experiments that covered an equivalent region. In other words, an analyst performing the same RSM study without a surrogate search capability would probably establish a smaller second order design region than ours. This would result in the need for another gradient search, or ridge analysis and another set of experiments over another second order design to reach the same results we achieved in a single set of second order experiments. Hence, we conclude that the surrogate search methodology reduced the total number of simulation replications performed by a factor of a second order design in order to reach satisfactory results.

7.4 Conclusion

In this chapter, we have demonstrated the effectiveness of the surrogate search methodology for two different simulation models. In both a simple study and a complex one, we have shown that the surrogate search methodology is fully integrated within the context of a simulation RSM study. It provides a tool to the analyst to not only reduce study times but to also enhance the analyst's knowledge of the system under study. Further, the flexibility of the method to adapt to different situations is made apparent via the demonstrations. For simulation models that are used on a regular basis, such as AFM, the cost of developing and validating an analytical model can be repaid as the surrogate search method is applied in several different studies.

VIII. Summary and Recommendations

8.1 Overview

This dissertation makes significant contributions towards the synergistic use of both analytical and simulation models to reduce the time required to complete a simulation study. The particular advancements in the field of using both types of models in concert are summarized below, followed by suggestions for future research.

8.2 Contributions

The significant contributions achieved by this research are summarized below.

8.2.1 ACV Monte Carlo Method. As mentioned before, previous researchers [48,49,53,54] have all reported unacceptable levels of bias when using ACV controlled estimators. This bias is caused by the necessity to evaluate the expected value of the ACV given the distribution of the input random variables used to produce the ACV [49]. The development of an efficient method to resolve the bias problem using a general Monte Carlo sampling technique makes a significant contribution to the field of simulation variance reduction and the synergistic use of analytical and simulation models. A summary of this work has been accepted for publication in the peer reviewed, archival journal, *IIE Transactions* [24].

8.2.2 ACV Monte Carlo Method with Incomplete Distributional Knowledge. The ACV Monte Carlo method described above relies upon the knowledge of the means and variances, as well as the approximate distributions, of the inputs to the analytical model. In this dissertation, we extend the range of permissible simulation models where the ACV method can be applied by describing alternative methods of generating the appropriate random vectors used to approximate the ACV mean. This research significantly advances the field of ACV application.

8.2.3 Surrogate Search Method. A new method for performing, and justifying, searches of a simulation design region using an analytical model is developed in this dissertation. The justification for the method is adapted from classic simulation model validation and verification. We demonstrate how the validation and surrogate search method developed in this research is fully integrated within a simulation study by analyzing the results of the ACV method. The new method and successful application of the method using two different simulation models advances the field of ACV's and of the synergistic use of both analytical and simulation models. To complete this research a useful analytical model of the Air Force simulation model MASS is developed—a first in the field. Additionally, two new performance measures for assessing the efficient movement of cargo within the airlift system are developed.

8.3 Recommendations for Future Research

Related topics, which could not be completed within this research, are described below.

The first research topic is a further development of the ACV method with incomplete distributional knowledge. The methods described in this research could be tested on simulation models with more variables and/or more variance in the output statistics. Further research into non-parametric methods of distribution sampling could also be explored.

Another area of possible research is the development of a *surrogate screen* method. In this method, the goal is to reduce the number of factors in a simulation experimental design by again using an analytical model in place of the simulation model. A proposed method could be an extension of the surrogate search method and could certainly be used in conjunction with the surrogate search method.

More research into the application of the surrogate search method could also be performed. For example, a likely area of fruitful research is developing methods that work when problems occur when attempting to apply the method. We touched upon some of the difficulties that can occur when applying the method, but certainly more work in this area could be accomplished. Also, research into further uses of the method beyond RSM studies could certainly be performed.

Finally research into formulating criteria for the types of simulation models and the conditions that are necessary for successful variance reduction using the ACV method, and thus successful surrogate searches, could be performed. Successful completion of this type of research would certainly be beneficial to simulation analysts.

Appendix A. Glossary of Acronyms and Abbreviations

AB Air Base

ACR aircraft cargo ratio

ACV analytical control variate

AFB Air Force Base

AMC Air Mobility Command

AFM Airlift Flow Model

BCMP Baskett, Chandy, Muntz, and Palacios (defines a type of queueing network)

BRACE Base Resource and Airfield Capability Evaluation

CCD central composite design

CPU central processing unit

CRAF Civil Reserve Air Fleet

CV control variate

D delay station

DOD Department of Defense

DOE design of experiment

ECA early cargo per aircraft

ECV external control variate

FCFS first come, first served

IAP international airport

IID independently and identically distributed

IV&V independent validation and verification

JFK John F. Kennedy

LCFS last come, first served

LOX liquid oxygen

MASS Mobility Analysis Support System

MOG maximum on the ground

MSE mean square error

MSPE mean square error for pure error

MVA mean value analysis

PC personal computer

PS processor sharing

RSM response surface methodology

SSE sum of square errors

SSPE sum of squares for pure error

SSPQ sum of squares for pure quadratic terms

SSR sum of squares, regression

TPFDD Time-Phased Force Deployment Data document

USAF United States Air Force

VRT variance reduction technique

Bibliography

1. Air Mobility Command. *Base Resource and Capability Estimator User's Manual Incomplete Draft*, May 1997.

2. R. Anonuevo and Barry L. Nelson. Automated estimation and variance reduction via control variates for infinite-horizon simulations. *Computers and Operations Research*, 15:447–456, 1988.

3. Athanassios N. Avramidis and James R. Wilson. A splitting scheme for control variates. *Operations Research Letters*, 14:187–198, 1993.

4. Osman Balci. How to assess the acceptability and credibility of simulation results. In *Proceedings of the 1989 Winter Simulation Conference*, 1989.

5. Osman Balci. Validation, verification, and testing techniques throughout the life cycle of a simulation study. *Annals of Operations Research*, 53:121–173, 1994.

6. Osman Balci and Robert Sargent. Some examples of simulation model validation using hypothesis testing. In *Proceedings of the 1982 Winter Simulation Conference*, 1982.

7. J. Banks, John S. Carson, and Barry L. Nelson. *Discrete-Event System Simulation.* Prentice-Hall, Englewood Cliffs, NJ, second edition, 1996.

8. Forest Baskett, K. Mani Chandy, Richard R. Muntz, and Fernando G. Palacios. Open, closed, and mixed networks of queues with different classes of customers. *Journal of the ACM*, 22:248–260, 1975.

9. Kenneth W. Bauer, Jr. and James R. Wilson. Control variate selection criteria. *Naval Research Logistics*, 39:307–321, 1992.

10. Kenneth W. Bauer, Jr. and James R. Wilson. Standardized routing variables: A new class of control variates. *Journal of Statistical Computation and Simulation*, 46:69–78, 1993.

11. Boeing. Mobility analysis support system (mass) migration. Technical Report 059D009, Defense Enterprise Integration Services, Joint Requirements Analysis and Integration Directorate, 1996.

12. George E. P. Box and Norman R. Draper. *Empirical Model-Building and Response Surfaces.* John Wiley & Sons, New York, 1987.

13. Steven C. Bruell and Giofranco Balbo. *Computational Algorithms for Closed Queueing Networks.* Elsevier North Holland, Inc., New York, 1980.

14. J. M. Burt, Jr., D. P. Gaver, and M. Perlas. Simple stochastic networks: Some problems and procedures. *Naval Research Logistics Quarterly*, 17:439–459, 1970.

15. John S. Carson. Convincing users of model's validity is challenging aspect of modeler's job. *Industrial Engineering*, 18:74–85, June 1986.

16. Adrian E. Conway and Nicolas D. Georganas. *Queueing Networks—Exact Computational Algorithms: A Unified Theory Based on Decomposition and Aggregation.* The MIT Press, Cambridge, MA, 1989.

17. Dennis C. Dietz. Mean value analysis of military airlift operations at an individual airfield. Submitted to Journal of Aircraft, May 1998.

18. Dennis C. Dietz and Chatherine M. Harmonosky. Application of a control variate technique to simulation analysis of aircraft sortie generation. IMSE Working Paper 89–109, Pennsylvania State University, 1989.

19. Dennis C. Dietz and Richard C. Jenkins. Analysis of aircraft sortie generation with the use of a fork-join queueing network model. *Naval Research Logistics*, 44:153–164, 1997.

20. Bradley Efron. Bootstrap methods—another look at the jackknife. *Annals of Statistics*, 7:1–26, 1979.

21. Bradley Efron. *The Jackknife, the Bootstrap and Other Resampling Plans.* Society for Industrial and Applied Mathematics, Philadelphia, 1982.

22. G. S. Fishman and P. J. Kiviat. The statistics of discrete-event simulation. *Simulation*, 10:185–195, 1968.

23. D. P. Gaver and G. S. Shedler. Control variable methods in the simulation of a model of a multiprogrammed computer system. *Naval Research Logistics Quarterly*, 18:435–450, 1971.

24. Thomas H. Irish, Dennis C. Dietz, and Kenneth W. Bauer Jr. Replicative use of an external analytical model in simulation variance reduction. Accepted for publication by IIE Transactions, July 1999.

25. K. C. Kapur and L.R. Lamberson. *Reliability in Engineering Design.* John Wiley and Sons, New York, 1977.

26. F. P. Kelly. Networks of queues and the method of stages. *Journal of Applied Probability*, 12:542–554, 1975.

27. A. I. Khuri and J. A. Cornell. *Response Surfaces: Designs and Analysis.* Marcel Dekker Inc., ASQC Quality Press, New York, 1987.

28. Leonard Kleinrock. *Queueing Systems, Volume1: Theory.* John Wiley & Sons, New York, 1975.

29. S. S. Lavenberg and M. Reiser. Stationary probabilities at arrival instants for closed queueing networks with multiple types of customers. *Journal of Applied Probability*, 17:1048–1061, 1980.

30. Stephen S. Lavenberg, Thomas L. Moeller, and Peter D. Welch. Statistical results on control variables with applications to queueing network simulation. *Operations Research*, 30:182–202, 1982.

31. Stephen S. Lavenberg and Peter D. Welch. A perspective on the use of control variables to increase the efficiency of monte carlo simulations. *Managment Science*, 27:322–335, 1981.

32. Averill M. Law and W. David Kelton. *Simulation Modeling and Analysis.* McGraw-Hill Book Company, New York, second edition, 1991.

33. Dave Merrill. Point paper on the mobility analysis support system prepared for the congressional budget office, October 1993.

34. Barry L. Nelson. On control variate estimation. *Computers and Operations Research*, 14:219–225, 1987.

35. Barry L. Nelson. Variance reduction for simulation practitioners. In *Proceedings of the 1987 Winter Simulation Conference*, pages 43–57, 1987.

36. Barry L. Nelson. Control variate remedies. *Operations Research*, 38:974–992, 1990.

37. John Neter and Michael H. Kutner. *Applied Linear Statistical Models.* Richard D. Irwin, INC, Burr Ridge, Illinois, third edition, 1990.

38. A. M. Porta Nova and James R. Wilson. Estimation of multiresponse simulation metamodels using control variates. *Management Science*, 35:1316–1333, 1989.

39. A. M. Porta Nova and James R. Wilson. Selecting control variates to estimate multiresponse simulation metamodels. *European Journal of Operational Research*, 71:80–94, 1993.

40. A. Alan Pritsker. *Introduction to Simulation and SLAM II.* John Wiley & Sons, New York, third edition, 1986.

41. P. Chandrasekhar Rao and Rajan Suri. Approximate queueing network models for closed fabrication/assembly systems. part i: Single level systems. *Production and Operations Management*, 3:244–275, 1994.

42. M. Reiser and S. S. Lavenberg. Mean-value analysis of closed multichain queueing networks. *Journal of the Association for Computing Machinery*, 27:313–322, 1980.

43. Sheldon M. Ross. *Introduction to Probability Models.* Academic Press, Inc., Boston, fifth edition, 1993.

44. R. Y. Rubinstein and R. Marcus. Efficiency of multivariate control variates in monte carlo simulation. *Operations Research*, 33:661–677, 1985.

45. Robert G. Sargent. Verifying and validating simulation models. In J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain, editors, *Proceedings of the 1996 Winter Simulation Conference*, 1996.

46. E. M. Scheuer and D. S. Stoller. On the generation of normal random vectors. *Technometrics*, 4:278–281, 1962.

47. Stewart Schlesinger et al. Terminology for model credibility. *Simulation*, 32:103–104, 1979.

48. Anthony P. Sharon. The effectiveness of jackson networks as control variates for queueing network simulation. Ms, Ohio State University, Columbus, Ohio, 1986.

49. Anthony P. Sharon and Barry L. Nelson. Analytic and external control variates for queueing network simulation. *Journal of the Operational Research Society*, 39:595–602, 1988.

50. M. S. Taylor and J. R. Thompson. A data based algorithm for the generation of random vectors. *Computational Statistics and Data Analysis*, 4:93–101, 1986.

51. Jeffery D. Tew and James R. Wilson. Validation of simulation analysis methods for the schruben-margolin correlation-induction strategy. *Operations Research*, 40:87–103, 1992.

52. James R. Thompson and Richard A. Tapia. *Nonparametric Function Estimation, Modeling, and Simulation.* Society for Industrial and Applied Mathematics, Philadelphia, 1990.

53. John J. Tomick. A comparison of control variates for queueing network simulation. Ms, Air Force Institute of Technology, Wright-Patterson AFB, Ohio, 1988.

54. John J. Tomick, Joseph R. Litko, and Kenneth W. Bauer Jr. A comparison of control variates for queueing network simulation. In *Proceedings of the 1989 Pittsburgh Simulation Conference*, 1989.

55. Sekhar Venkatraman and James R. Wilson. The efficiency of control variates in multiresponse simulation. *Operations Research Letters*, 5:37–42, 1986.

56. James R. Wilson and A. Alan B. Pritsker. Variance reduction in queueing simulation using generalized concomitant variables. *Journal of Statistical Computation and Simulation*, 19:129–153, 1984.

57. W. N. Yang and Barry L. Nelson. Multivariate estimation and variance reduction in terminating and steady-state simulation. In *Proceedings of the 1988 Winter Simulation Conference*, 1988.

Vita

Major Thomas H. Irish was born 20 February 1955 at Fort Leavenworth, Kansas. He graduated from Peoria High School (Illinois) in 1973 and enlisted in the United States Air Force in December 1976. Major Irish separated from the Air Force in 1980 to pursue his education. He graduated from California State University, Sacramento with the degree of Bachelor of Arts in Mathematics in 1985. Major Irish was a distinguished graduate of the United States Air Force Officer Training School and was commissioned in November 1985. His first assignment was to Fairchild AFB, Washington, as 92nd Field Maintenance Squadron Assistant Maintenance Supervisor. He has since served several tours as an Aircraft Maintenance Officer and as a Supply Officer. Major Irish entered the School of Engineering, Air Force Institute of Technology (AFIT) in August 1994 where he earned a Masters of Science degree in Operations Research in 1996 as a distinguished graduate. Upon graduation, Major Irish remained at AFIT to pursue a Doctor of Philosophy degree in Operations Research.

Permanent address: 105 LaVerne Ave.
Daphne, AL 36526

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
September 1999

3. REPORT TYPE AND DATES COVERED
Ph.D. Dissertation

4. TITLE AND SUBTITLE
EFFICIENT SIMULATION VIA VALIDATION AND APPLICATION OF AN EXTERNAL ANALYTICAL MODEL

5. FUNDING NUMBERS

6. AUTHOR(S)
Thomas H. Irish, Major, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology
2750 P Street
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/DS/ENS/99-01

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AMCSAF/XPY
402 Scott Dr. Unit 3L3
Scott AFB, IL 62225-5307
DSN: 576-2208

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This research makes significant contributions towards improving the efficiency of simulation studies using an external analytical model. The foundation for this research is the analytical control variate (ACV) method. The ACV method can produce significant variance reduction, but the resulting point estimate may exhibit bias. A Monte Carlo sampling method for resolving the bias problem is developed and demonstrated through a queueing network example. The method requires knowledge of the parameters and approximate distributions of the random variables used to produce the ACV. Often, some of these parameters or distributions are not known. Both parametric and non-parametric alternatives to the Monte Carlo method are explored for these cases.

Significant variance reduction using an ACV indicates that the outputs of both models are highly correlated. This relationship is exploited and a new methodology is developed for conducting searches of a simulation design space using an analytical model vice a simulation model. The justification for the new surrogate search method is based on validating the analytical model to the simulation model. The effectiveness of the method is demonstrated on two simulation models including the HQ AMC Mobility Analysis Support System (MASS) model.

14. SUBJECT TERMS
Simulation, Variance Reduction, Control Variates, Queueing Networks, Response Surface Methodogy

15. NUMBER OF PAGES
257

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL